ultimo theme

Extremely customizable Magento theme by Infortis

User Guide

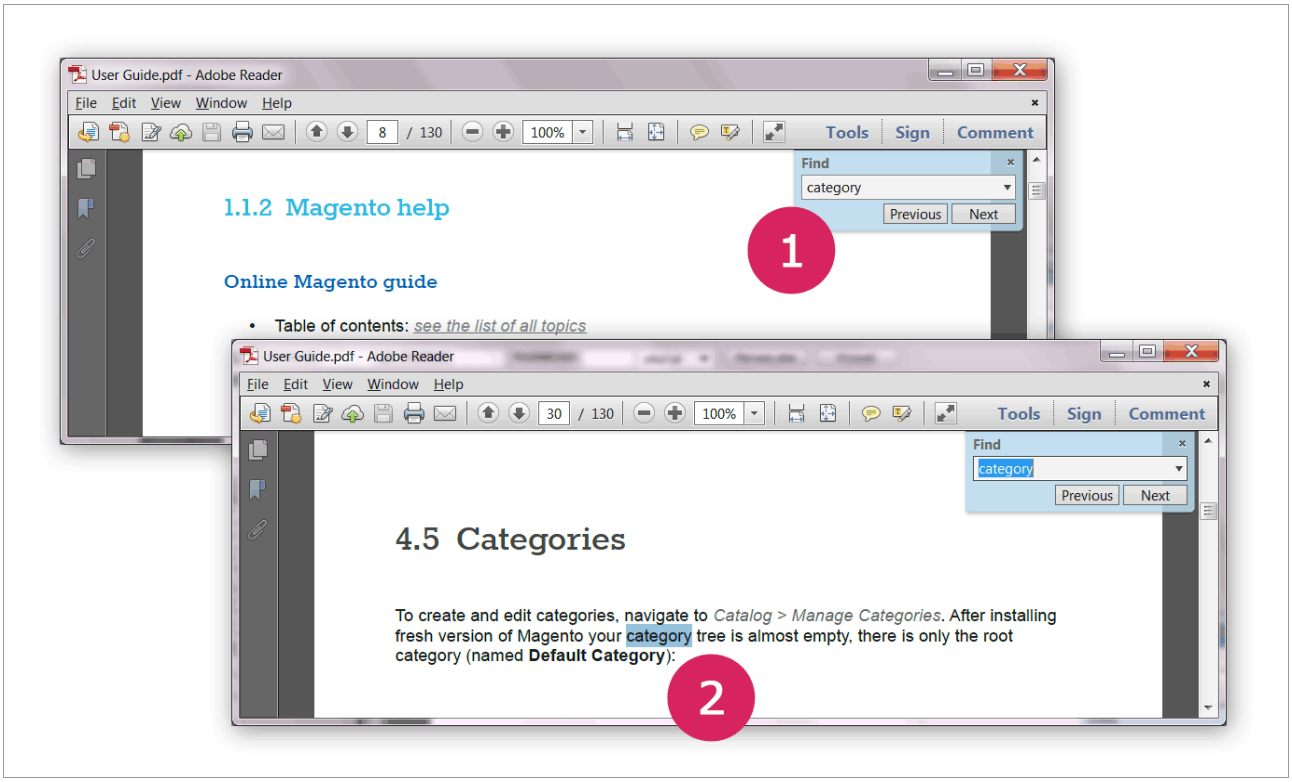
Copyright © 2012-2015 Infortis

All rights reserved

How to use this document

Please read this user guide carefully, it will help you eliminate most of potential problems with incorrect configuration of the theme and Magento.

Use the search tool in your PDF reader (click **CTRL+F** in most readers) to quickly find all the keywords which you are looking for:


User Guide.pdf - Adobe Reader
File Edit View Window Help
8 / 130
100%
Tools Sign Comment
Find
category
Previous Next
1.1.2 Magento help
Online Magento guide
Table of contents: see the list of all topics
1
User Guide.pdf - Adobe Reader
File Edit View Window Help
30 / 130
100%
Tools Sign Comment
Find
category
Previous Next
4.5 Categories
To create and edit categories, navigate to Catalog > Manage Categories. After installing fresh version of Magento your category tree is almost empty, there is only the root category (named Default Category):
2

This document is organized as follows:

- Chapters **1. FAQ + Quick Start** and **2. Help & Support** provide an overview of this theme, some basic information about support and links to many useful websites about Magento.

- Chapter **3. Installation** describes theme installation process.

- Chapter **4. Magento configuration** describe selected Magento features.

- Chapter **5. Customization** shows how to customize the theme. Subchapter **5.2** describes Magento themes structure and some basic concepts which you need to know when starting to work with Magento.

- Chapter **6. Theme features and elements** provides a description of selected elements of this theme, such as product sliders, home page, built-in static blocks, useful CSS classes etc.

- Chapters from **7.** to **13.** – specify a list of all theme settings available in the theme admin panel. **Each chapter refers to specific section of theme admin panel.**

- Chapter **14. Troubleshooting** describes the most common problems which you may encounter after installation or during theme and Magento configuration.

All chapters are listed in the **Table of Contents** below.

Table of Contents

1. FAQ + Quick Start....10
   1.1 Help & support....10
   1.2 License....11
   1.3 Important things you need to know before you start using Magento....12
   1.4 Installation....12
   1.5 After installation....12
   1.6 Fresh Magento installation....14
   1.7 Theme customization....14
2. Help & Support....15
   2.1 References....15
      2.1.1 Magento installation....15
      2.1.2 Magento help....16
   2.2 Support policy....18
   2.3 Third-party extensions....20
3. Installation....22
   3.1 Installation of demo store....22
   3.2 Installation for live Magento store....23
      3.2.1 Prepare your Magento....23
      3.2.2 Install....25
      3.2.3 Import sample blocks and pages....28
   3.3 Problems after installation....29
4. Magento configuration....31
   4.1 Caching system....32
      4.1.1 Flush cache....32
      4.1.2 Disable cache....33
      4.1.3 Enable cache....33
      4.1.4 Refresh cache....34
   4.2 Index management....35
   4.3 How to enable the theme....36
   4.4 Logo....38
   4.5 Categories....39
      4.5.1 Category info....42
      4.5.2 Landing pages....42
      4.5.3 Depth of the menu....42
      4.5.4 Layered Navigation ("Shop By" filters)....42
      4.5.5 Categories inside Layered Navigation....43
   4.6 Products....44
      4.6.1 Position of: Add to Cart button and product options....44
      4.6.2 "Only X left" stock indicator....45
   4.7 Attributes....46
      4.7.1 Attribute Management....46
      4.7.2 Attribute Sets....49
   4.8 Content Management System (CMS)....51
      4.8.1 Add blocks to selected pages....51

4.8.2 Static blocks....54
4.8.3 CMS markup tags....55
4.8.4 WYSIWYG editor....56
4.9 Translation / Localization....57
4.9.1 Interface translation....57
4.9.2 Language flag....58
5. Customization....59
5.1 How to modify the theme....60
5.1.1 Customize images....60
5.1.2 Override default CSS styles....61
5.1.3 Modify template files....63
5.1.4 Create custom sub-theme....64
5.2 Magento theme customization....65
5.2.1 Design packages....65
5.2.2 Theme structure....66
5.2.3 Fallback mechanism....67
5.2.4 Examples of custom sub-theme....69
5.2.5 Summary....72
6. Theme features and elements....73
6.1 Home page....74
6.1.1 Home page layout....75
6.1.2 Home page content....76
6.1.3 Home page blocks....77
6.1.3.A Image slideshow....77
6.1.3.B Banners....77
6.1.3.C Products grid....79
6.1.4 Featured Products Slider....80
6.1.5 New Products Slider....85
6.2 Built-in CMS blocks....89
6.2.1 Header....91
6.2.2 Footer....93
6.2.3 Category view....95
6.2.4 Product page....96
6.2.5 Home page....99
6.2.6 Shopping cart....100
6.2.7 Cart drop-down....100
6.2.8 One page checkout....100
6.2.9 Left column....100
6.2.10 Right column....100
6.2.11 CMS pages....101
6.3 Pages and elements....102
6.3.1 Product page....102
6.3.1.A Social bookmarks (AddThis)....102
6.3.2 Drop-down cart....102
6.4 Grid system....103
6.4.1 Simple grid examples....103
6.4.2 Multiple rows....105
6.4.3 Grid on mobile devices....105
6.5 CMS components....107
6.5.1 CSS utility classes....107
6.5.1.A Hide elements for specific screen resolution....107
6.5.2 Collapsible blocks....108
6.5.3 Icons....109
6.5.3.A Font icons....109
6.5.3.B Icons with background color (iconbox)....111
6.5.3.C Social icons....113
6.5.3.D Custom image icons....114
6.5.3.E Letters, numbers and other characters as icons....115
6.5.4 Text with icon....116

6.5.5 Captions for Slideshow....119
6.6 Theme customization....121
7. Theme settings....122
7.1 Header....123
7.1.1 Header structure....123
7.1.2 Header configuration options....125
7.2 Category View....128
7.3 Category View (Grid Mode)....130
7.4 Category View (List Mode)....133
7.5 Product Page....134
7.5.1 Product page structure....134
7.5.2 Product page block positions....143
7.5.3 Product page configuration options....144
7.6 Product Labels....152
7.7 Footer....154
7.8 Product sliders....155
7.9 Default Magento blocks....156
7.10 Disable Sidebar Blocks (On Home Page)....157
7.11 Customization Settings....158
7.12 Data Import....159
8. Theme Design....165
8.1 Colors....166
8.2 Font....167
8.3 Effects....167
8.4 Page....167
8.5 Header....168
8.6 Header – Top....169
8.7 Header – Primary....169
8.8 Main Menu....169
8.9 Main....169
8.10 Category View....169
8.11 Product Page....169
8.12 Slideshow....170
8.13 Footer....170
8.14 Footer – Top....170
8.15 Footer – Primary....170
8.16 Footer – Secondary....170
8.17 Footer – Bottom....171
9. Theme Layout....172
9.1 Responsive Layout....173
9.2 Disable responsive layout....173
10. Zoom....175
10.1 Zoom configuration....175
10.1.1 General....175
10.1.2 Image Size....177
10.1.3 Thumbnails....178
10.1.4 Lightbox....179
11. Slideshow....180

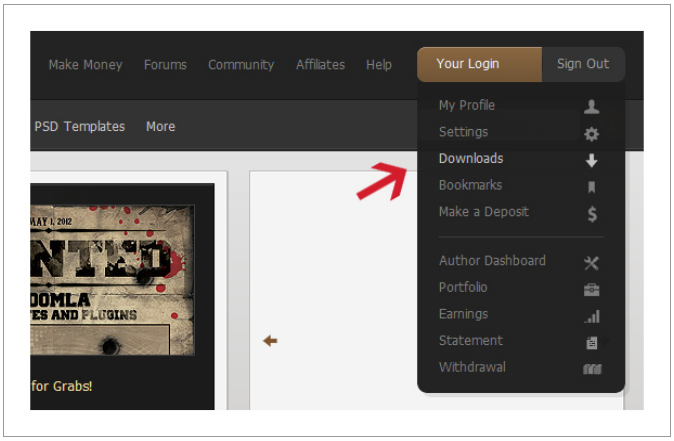
11.1 How to display a slideshow on any page....180
11.1.1 List of all parameters....181
11.1.2 Full width slideshow on home page....181
11.2 Static blocks as slides....183
11.2.1 Size of slides and banners....183
11.2.2 Clickable images....185
11.3 Slideshow configuration....186
11.3.1 General....186
11.3.2 Banners....188
11.3.3 Navigation....189
12. Brands....190
12.1 Basic information....190
12.1.1 Brand attribute....190
12.1.2 Brand images....191
12.1.3 Brand names and image file names....191
12.2 Brands configuration....193
12.2.1 General....193
12.2.2 Brand on Product Page....198
12.2.3 Brand List....199
12.2.4 Brand Slider....200
13. Menu....203
13.1 Menu configuration....204
13.1.1 General....204
13.1.2 Menu Bar....205
13.1.3 Mobile Menu....206
13.1.4 Sidebar Menu....207
13.1.5 Category Labels....211
13.2 Drop-down boxes....212
13.2.1 Drop-down box configuration options....212
13.2.2 Submenu types....216
13.2.3 Category blocks....219
13.2.3.A Category blocks in mobile menu....221
13.2.4 Sample content for category blocks....223
13.3 Custom links in the menu....225
13.4 Custom drop-down boxes in the menu....228
14. Troubleshooting....230
14.1 General issues after installation....230
14.1.1 Some features don't work or give errors....230
14.1.2 Blank page or 404 not found – when trying to open theme admin pages....231
14.1.3 There is no theme admin tab in the admin panel, or I get a blank page, or Access Denied error 231
14.1.4 File permissions....231
14.1.5 Theme features don't work after installation or upgrade....231
14.1.6 Fatal error: Class 'Mage_Default_Helper_Data' not found....232
14.1.7 "Package with this name does not exist" message is displayed when trying to enable the theme 232
14.1.8 After theme installation the store gives blank page or an error....233
14.1.9 Tabs, sliders or other interactive elements don't work....233
14.1.10 Images are not displayed....233
14.1.11 Sliders: previous/next buttons don't work in some browsers....233
14.1.12 Slideshow is not displayed on the home page....234
14.2 Configuration....235
14.2.1 Theme configuration changes do not appear on the frontend....235
14.2.2 "Failed generating CSS file" message after saving theme configuration....235
14.3 Third-party extensions....236
14.3.1 One of my extensions doesn't work with this theme....236
14.3.2 One of the features stopped working after third-party extension was installed....236

Introduction

Ultimo is a Magento theme with advanced admin module, it's extremely customizable and fully responsive. It is suitable for every type of store. It is a separate design package (not a sub-theme of Magento “default” theme) so you can easily create your own sub-themes.

Rate this theme

Don't forget to rate this theme on *ThemeForest* – it's good to know what others think about our work. You can rate it in the download area of your account on ThemeForest:

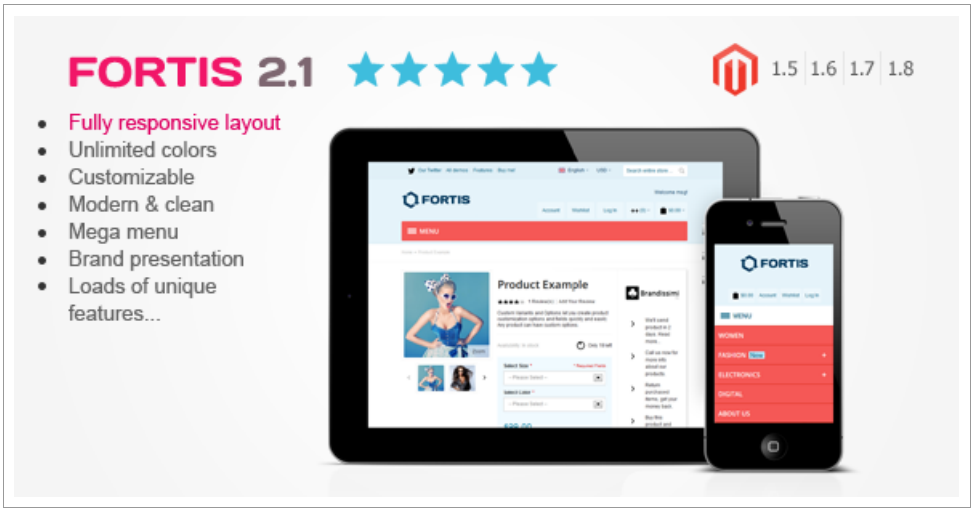


More info

To find out more about theme updates, new features and about our upcoming themes, you can follow us on Twitter and on ThemeForest:

1. FAQ + Quick Start

See also our Fortis theme for Magento (click the image):



1.1 Help & support

1. **Theme support**

   Please read this user guide carefully, it will help you eliminate most of potential problems with incorrect configuration of the theme and Magento. If you don't find the answer to your questions, please refer to this chapter for more information about support policy: **2.2 Support policy**.

2. **Magento support**

   Magento configuration, installation, maintenance, customization etc. is beyond the scope of our support. But since Magento is very popular platform you can find tons of useful tutorials and articles on the web. Here are some resources for start: **2. Help & Support**.

1.2 License

With **ThemeForest regular license** you are licensed to use this theme to create one single End Product (the final website customized with your content) for yourself or for one client. In short:

What is allowed with single regular license:

1. **You can create one website** for yourself or for your client and you can transfer that single website to your client **for any fee**. This license is then transferred to your client.

2. To sell the same website to another client, you will need to purchase **another regular license**.

3. You can install the theme on your **test server** for testing/development purposes (that installation shouldn't be available to the public).

4. You can modify or manipulate the theme, you can **combine it with other works** to create the End Product.

5. Theme **updates** for that single End Product (website) are free.

What is not allowed:

1. With single regular license it is not allowed to create **multiple websites**. For multiple websites you will need multiple regular licenses.

2. **Multi-store Magento installations** are not allowed with single regular license. Each regular license entitles you to use the theme in only one store (store/domain/subdomain). For multiple stores/domains/subdomains you will need multiple regular licenses.

For more information about licenses please refer to ThemeForest:

- License FAQ: *http://themeforest.net/licenses/faq*
- License comparison table: *http://themeforest.net/licenses*
- Regular license details: *http://themeforest.net/licenses/regular*
- Extended license details: *http://themeforest.net/licenses/extended*

1.3 Important things you need to know before you start using Magento

1. ### Disable Magento cache

   When developing your store you should completely **disable Magento cache**. Enable it after you finish configuring your store: **4.1 Caching system**.

2. ### Reindex Magento

   After making changes to your store such as editing products, adding categories etc. you need to reindex the data: **4.2 Index management**.

1.4 Installation

To install the theme you have two options:

- If you already have a Magento store and want to apply the theme to your store, follow the instructions in **3.2 Installation for live Magento store**.

- If you want to test the theme and check how the live demo was configured, you can install demo containing the sample data and settings from the live demo. Follow the instructions in **3.1 Installation of demo store**.

If you encounter any problems after installation, refer to chapter **14. Troubleshooting**.

1.5 After installation

To start using the theme after installation you need to take a few quick steps to make sure all is configured properly. Also, not all of the features are enabled by default.

1. ### Enable the theme in a correct way

   Enable the theme in a correct way which is described in the installation instructions: **3.2.2 Install**.

> **Important:**
>
> do not enable the theme in `System > Design`. It should only be enabled in `System > Configuration > Design`.

In Magento theme can be enabled for the entire store, but you can also set different themes for different parts of the store (even for selected categories, products or CMS pages). If you have already used some other theme in your store, you will need to remove all theme overrides from categories, products and CMS pages. See for more details: **4.3 How to enable the theme**.

2. ### Disable cache

   In order to be able to see theme configuration changes immediately, you should completely disable the cache. You can enable it after you finish configuring your store. See for more details: **4.1 Caching system**.

3. ### Import configuration settings (optional)

   After the theme was installed and enabled, you can import theme configuration settings (colors, font, page width etc.) from one of the demo stores. For more details refer to **7.12 Data Import**. Of course, **configuration import is optional** and you can as well leave the default configuration (by default theme looks the same as demo 1).

> **Note:**
>
> if you decide to import configuration from one of the demo stores, it is recommended to import to the **"Default Config"** scope.

4. ### Logo

   Upload your store logo: **4.4 Logo**.

5. ### Home page

   Configure your home page: **6.1 Home page**.

6. ### Home page slideshow

   Configure the home page slideshow: **11. Slideshow**. Each slide is a static block

which can contain any custom images or HTML. After blocks import, your slideshow will have a few sample slides filled with some dummy images and captions. To edit captions, see this chapter for details: **6.5.5 Captions for Slideshow**.

7. ### Position of "Add to cart" button and product options

   Configure position of the **Add to cart** button and product options box on product page: **4.6.1 Position of: Add to Cart button and product options**.

> **Note:**
>
> you may find these Magento tutorials and documentation useful while configuring your store: **2.1 References**

1.6 Fresh Magento installation

If you have just installed a fresh copy of Magento without any sample data, here's where you should start:

1. Disable Magento cache. In order to be able to see configuration changes immediately, you should completely disable the cache. You can enable it after you finish configuring your store. See for more details: **4.1 Caching system**.

2. Create some categories in `Catalog > Manage Categories` – by default the category tree is completely empty. See for more details: **4.5 Categories**.

3. Create some products in `Catalog > Manage Products`.

4. Reindex Magento after creating categories and products: **4.2 Index management**. You will also need to reindex from time to time after making changes in Magento catalog (categories, products, product attributes, attribute sets etc.).

1.7 Theme customization

There are many ways to customize the theme, for more information refer to chapter **5. Customization**. Customization/modification is beyond the scope of our support.

2. Help & Support

2.1 References

You may find these sites useful while installing and configuring your Magento store:

2.1.1 Magento installation

- **Magento FAQ:**

  http://www.magentocommerce.com/product/faq

- **System Requirements:**

  http://www.magentocommerce.com/system-requirements/

- **Server Compatibility Check:**

  http://www.magentocommerce.com/knowledge-base/entry/how-do-i-know-if-my-server-is-compatible-with-magento

- **Installation Guide:**

  http://www.magentocommerce.com/knowledge-base/entry/magento-installation-guide

2.1.2 Magento help

Online Magento guide

- Table of contents: *click to see the list of all topics*
- Selected topics:
    1. *Categories, products and attributes*
    2. *Customers*
    3. *Payment, shipping and taxes*
    4. *Promotions and marketing*

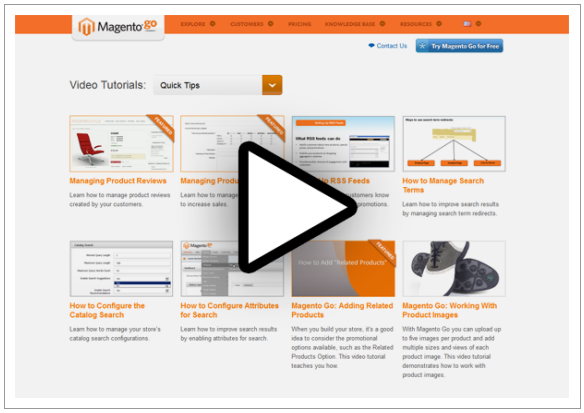
Magento Go Knowledge Base

http://go.magento.com/support/kb

> **Note:**
>
> ***Magento Go*** differs from ***Magento Community Edition*** but most of the functionality is the same so this knowledge base might be very useful.

Magento Go Video Tutorials

http://go.magento.com/video-tutorials


Video Tutorials:
Quick Tips
Managing Product Reviews
How to Manage Search Terms
How to Configure the Catalog Search
How to Configure Attributes for Search
Magento Go: Adding Related Products
Magento Go: Working With Product Images

Magento Wiki

http://www.magentocommerce.com/wiki/

Magento Forum

http://www.magentocommerce.com/boards/

stackoverflow.com (Q&A)

http://stackoverflow.com/questions/tagged/magento

Magento Stack Exchange (Q&A)

http://magento.stackexchange.com/

2.2 Support policy

This user guide was created to help you quickly configure the theme – please read it carefully, it will help you eliminate most of potential problems with incorrect configuration.

> **Important:**
>
> please note that as authors of this theme we are able to provide support only for the features which we created and for the issues related strictly to this theme. We do not provide support for Magento configuration, installation, maintenance, customization etc.

Theme support

If you have found any bugs or have some other problems with this theme, please refer to chapter **14. Troubleshooting**. If the problem is not covered there, you can contact us in the comment section on ThemeForest or through the contact form on our profile page on ThemeForest. The most up-to-date information about support can always be found at:

http://themeforest.net/item/ultimo-fluid-responsive-magento-theme/3231798/support

We will respond as soon as possible (within 24 – 48 hours, usually faster). Please make sure you take a look at the available resources before submitting a support request.

Questions about Magento

Elements such as *category management*, *product management*, *checkout process*, *payment methods* and many others are standard Magento functionalities (not related to the theme) and are beyond the scope of our support. How to get Magento support:

1. If you have some questions about Magento itself, please refer to the following websites, almost every detail of Magento configuration was discussed there so you will find answers for most of your questions:
    - ***Magento forum*** – `http://www.magentocommerce.com/boards/`
    - ***Magento Stack Exchange*** – `http://magento.stackexchange.com/`

- ***Stackoverflow.com*** – `http://stackoverflow.com/questions/tagged/magento/`

2. The best place to start searching is chapter **4. Magento configuration** in this user guide. It covers some of the basic Magento elements.

3. In the ***Online Magento Guide*** you can find description of all the basic Magento settings and many other useful information.

4. You might also want to check the other sites referenced in **2.1.2 Magento help** such as **Magento Go Knowledge Base** or **Magento Go Video Tutorials**.

Theme customization/modification support

Customization/modification of Magento themes is a very large subject and can be very complex. Due to that we're unable to provide support related to theme customization. For more information about how you can customize this theme, refer to **5. Customization**.

2.3 Third-party extensions

This theme may not be compatible with some third-party extensions. It's impossible for theme authors to make the theme compatible by default with all existing extensions because there are thousands of available Magento extensions. Only the "default" Magento theme is compatible with all third-party extensions, because all extensions are designed to work with it. But authors of the extension should provide detailed instruction about how to integrate the extension with custom themes, since nobody really uses "default" theme, everybody uses custom themes.

Generally all extensions can work with all themes but they sometimes have to be integrated manually. That's because sometimes the extension and the theme override the same Magento template file so there can be a conflict. In such cases you need to customize the theme and merge two conflicting template files into one template file.

This isn't something specific to our themes, this is general rule for all Magento themes and extensions.

Support

Unfortunately third-party extensions are beyond the scope of our support, please contact the author of the extension if needed. Author of the extension should provide detailed instructions about how to integrate the extension with any custom theme.

How to install

To work with this theme, an extension should be installed in theme's directories, not in the "default" Magento theme directories. To be more specific: extension should be installed in the main directories of the default theme in the Ultimo design package. Here are the main directories of Ultimo:

- Template files:

  `app/design/frontend/ultimo/default`

- Skin files:

  `skin/frontend/ultimo/default`

where `ultimo` is the name of the design package, and `default` is the name of the default theme in the package.

In many cases you only need to upload the extension to the theme's directories. Sometimes you need to edit layout files to "tell" the theme where it should display extension's output. If you are not familiar with Magento, you should ask the author of the extension for more information. If you feel that you might have troubles with installing extensions, we advise that you order professional installation services (unfortunately we don't offer this type of services). You can find professional help on sites like, for example, *freelancer.com* or *freelanceswitch.com* .

3. Installation

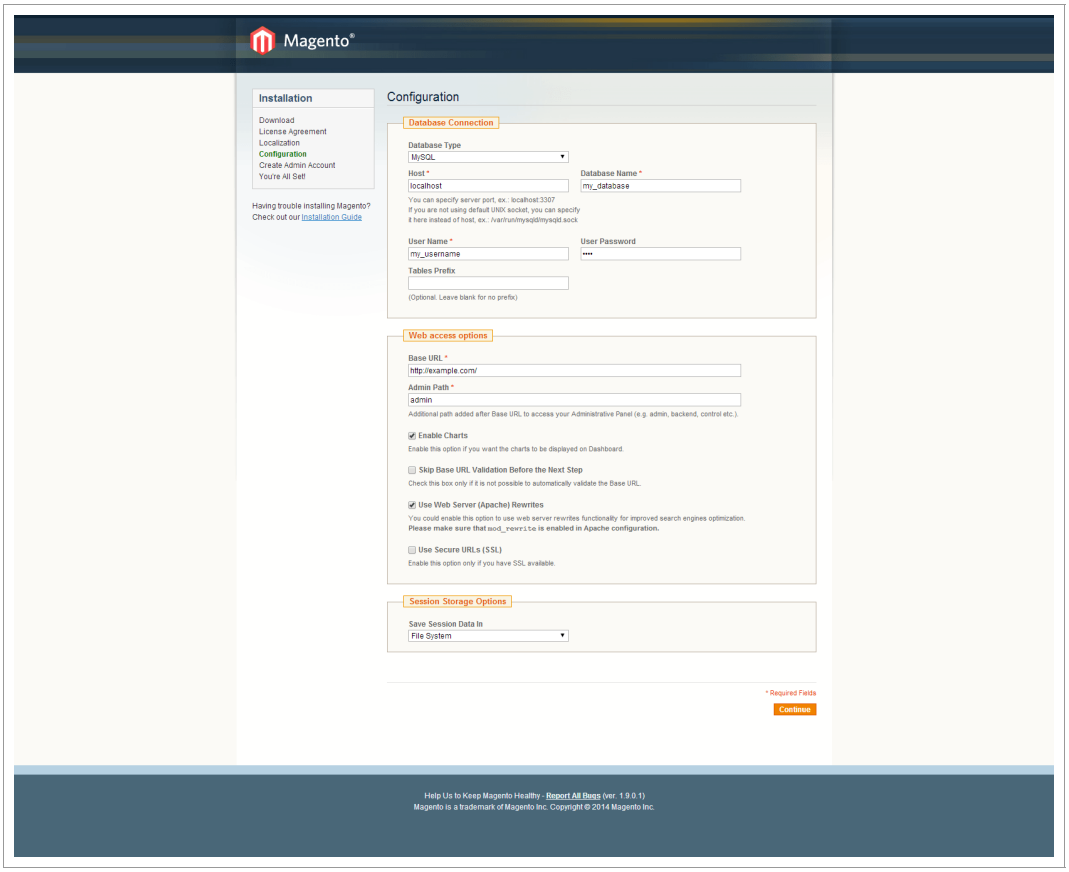
To install the theme you have two options:

- If you already have a Magento store and want to apply the theme to that store, follow the instructions in **3.2 Installation for live Magento store**.
- Or if you want to test the theme and check how the live demo was configured, you can install demo containing all the sample data and settings from our live demo. Follow the instructions in **3.1 Installation of demo store**.

3.1 Installation of demo store

To install demo store with all the sample data (similar to live demo of Ultimo), follow the instruction below. These instructions assume you are in the installation phase with your Magento store. If you already have a working Magento store and database – go to **3.2 Installation for live Magento store**.

- Navigate inside `Quick Start Demo Package`, extract the package on your computer and navigate inside.
- Create a new empty database, find `demo_database.sql.gz` inside the package and import it into your newly created database (using your database management tool such as *phpMyAdmin* or whatever you use), right before Magento installation. The demo data must be imported before running the Magento Setup Wizard.
- Now you can install Magento: using your FTP client, upload `Magento.zip` file to your server (to the folder where Magento can be installed) and extract it there.
- Open a web browser and navigate to your website (to the folder where you extracted `Magento.zip`) to load the Magento Setup Wizard.
- Follow the installation instructions step by step just like in case of a standard Magento installation process. During the installation process, enable **Web Server Rewrites** functionality for your Magento (it controls whether Magento will automatically generate search engine friendly URLs for products and categories). It may look like on the following screenshot – option `Use Web Server (Apache) Rewrites`:


Magento
Installation
Download
License Agreement
Localization
Configuration
Create Admin Account
You're All Set!
Having trouble installing Magento?
Check out our Installation Guide
Configuration
Database Connection
Database Type
MySQL
Host *
localhost
Database Name *
my_database
You can specify server port, ex.: localhost:3307
If you are not using default UNIX socket, you can specify
it here instead of host, ex.: /var/run/mysqld/mysqld.sock
User Name *
my_username
User Password
Tables Prefix
(Optional. Leave blank for no prefix)
Web access options
Base URL *
http://example.com/
Admin Path *
admin
Additional path added after Base URL to access your Administrative Panel (e.g. admin, backend, control etc.).
Enable Charts
Enable this option if you want the charts to be displayed on Dashboard.
Skip Base URL Validation Before the Next Step
Check this box only if it is not possible to automatically validate the Base URL.
Use Web Server (Apache) Rewrites
You could enable this option to use web server rewrites functionality for improved search engines optimization.
Please make sure that mod_rewrite is enabled in Apache configuration.
Use Secure URLs (SSL)
Enable this option only if you have SSL available.
Session Storage Options
Save Session Data In
File System
* Required Fields
Continue
Help Us to Keep Magento Healthy - Report All Bugs (ver. 1.9.0.1)
Magento is a trademark of Magento Inc. Copyright © 2014 Magento Inc.

- When you're ready to set up your own store, you should install Magento again without the sample data in order to start clean. Using sample data as a base for real store is not recommended.

3.2 Installation for live Magento store

If you already have a Magento store and want to apply the theme, follow the instruction below. To install the theme you only need to upload all the theme files to your server and then enable the theme in the admin panel. But to avoid unexpected problems after installation, you should first prepare your Magento. Please strictly follow the steps described in this guide:

3.2.1 Prepare your Magento

1. It is recommended to install the theme on a **test server** before you install it on a live store.

2. **Backup Magento** files and the store database.

> **Important:**
>
> creating backup before installation of any themes or extensions in Magento is extremely important, especially if you are working on a live store.

3. **Disable compilation** mode in `System > Tools > Compilation`.

4. **Disable all cache systems** which you have in your Magento:

   4.a. Magento's cache (refer to **4.1.2 Disable cache** for details)

   4.b. full page cache and caching modules for Magento (such as Speedster)

   4.c. any additional cache on your server, PHP cache engines, APC, etc.

> **Important:**
>
> when developing your Magento store, in order to be able to see changes immediately, you should completely disable the cache. You can enable it after you finish configuring your store.

5. Disable **Merge JavaScript Files** and **Merge CSS Files** in `System > Configuration > Developer`. You can enable this feature after installation.

6. Remove all possible custom modifications of the **Magento's "base" theme**. Ultimo (the same as any other Magento theme) relies on "base" theme, so any odifications of the "base" theme can change the default behavior of Ultimo and break some functionality. **You should never edit "base" theme's files**. Custom modifications should be always made via custom sub-themes.

7. **Log out** from Magento admin panel.

> **Important:**
>
> do not just close the browser window: you need to click the **Log Out** link to refresh the access control system.

8. Before you install, make sure your store is disabled during the installation process. Put Magento in the **maintenance mode**. To do this you can create a simple empty file named `maintenance.flag` in your Magento's root directory. Magento will be offline after that (and you will not be able to log in to the admin panel). You can

remove the file after you finish.

3.2.2 Install

Before you can install the theme you need to prepare your Magento, so please do not omit the previous step: **3.2.1 Prepare your Magento**.

9. Extract the theme package on your computer and navigate inside `Theme Files` directory.

10. Using your FTP client upload `Theme.zip` file into the root directory of your Magento installation (the root directory of Magento is the folder that contains files, such as `index.php` and `get.php` and the directories `app`, `js`, `lib`, `skin` and more).

11. Unzip `Theme.zip` file – it contains four directories: `app`, `js`, `media`, `skin`. Magento already contains the same directories - **do not remove those existing directories**. Directories extracted from `Theme.zip` have to be merged with existing directories of Magento. Approve, if your system asks you whether you want to merge directories.

> **Note:**
>
> this will not overwrite any core files of Magento, this will only add the theme to your Magento. So if you see a message that those four directories already exist on the server, you can confirm that you want to merge directories.

In case you can't unzip files on your server:
Alternatively, you can unzip `Theme.zip` archive on your computer, and upload its content (four directories: `app`, `js`, `media`, `skin`) to your Magento's root directory. But note that files can get broken if you send them via FTP without zipping. So **it is highly recommended to upload zipped files**.

12. **Install theme patches for specific versions of Magento**. In this case patch doesn't fix anything, it only adds theme files which are required in specific versions of Magento. Please note that no patch is needed for Magento 1.7.x.

    12.a. ***NEW:*** *Omit this step if you are not using Magento 1.9.1.0*

    This step is required only if you are using Magento **1.9.1.0**. In this step you will install files which are required only in Magento 1.9.1.0. Navigate inside `Theme (patch for Magento 1.9.1.0)` directory. Using your FTP client upload content of `Theme (patch for Magento 1.9.1.0)` directory to Magento's root directory. Overwrite existing files.

12.b. Omit this step if you are not using Magento 1.8.1.0, 1.9.0.0 or 1.9.0.1

This step is required only if you are using Magento **1.8.1.0**, **1.9.0.0** or **1.9.0.1**. In this step you will install files which are required only in those versions of Magento. Navigate inside `Theme (patch for Magento 1.8.1.0, 1.9.0.0, 1.9.0.1)` directory. Using your FTP client upload content of `Theme (patch for Magento 1.8.1.0, 1.9.0.0, 1.9.0.1)` directory to Magento's root directory. Overwrite existing files.

12.c. Omit this step if you are not using Magento 1.8.0.0

This step is required only if you are using Magento **1.8.0.0**. In this step you will install files which are required only in Magento 1.8.0.0. Navigate inside `Theme (patch for Magento 1.8.0.0)` directory. Using your FTP client upload content of `Theme (patch for Magento 1.8.0.0)` directory to Magento's root directory. Overwrite existing files.

12.d. Omit this step if you are not using Magento 1.6.x

This step is required only if you are using Magento **1.6.x**. In this step you will install files which are required only in Magento 1.6.x. Navigate inside `Theme (patch for Magento 1.6.x)` directory. Using your FTP client upload content of `Theme (patch for Magento 1.6.x)` directory to Magento's root directory. Overwrite existing files.

12.e. Omit this step if you are not using Magento 1.5.x

This step is required only if you are using Magento **1.5.x**. In this step you will install files which are required only in Magento 1.5.x. Navigate inside `Theme (patch for Magento 1.5.x)` directory. Using your FTP client upload content of `Theme (patch for Magento 1.5.x)` directory to your Magento root directory. Overwrite existing files.

13. **Set correct file permissions** for all the theme files which you uploaded to your server.

> **Note:**
>
> theme files should have the same permissions as other Magento files. Otherwise Magento will not have access to the theme files, images etc. File permissions may differ depending on server. In case of any doubts about the permissions, please contact your hosting provider. Refer to this article for more details: ***Magento Filesystem Permissions***

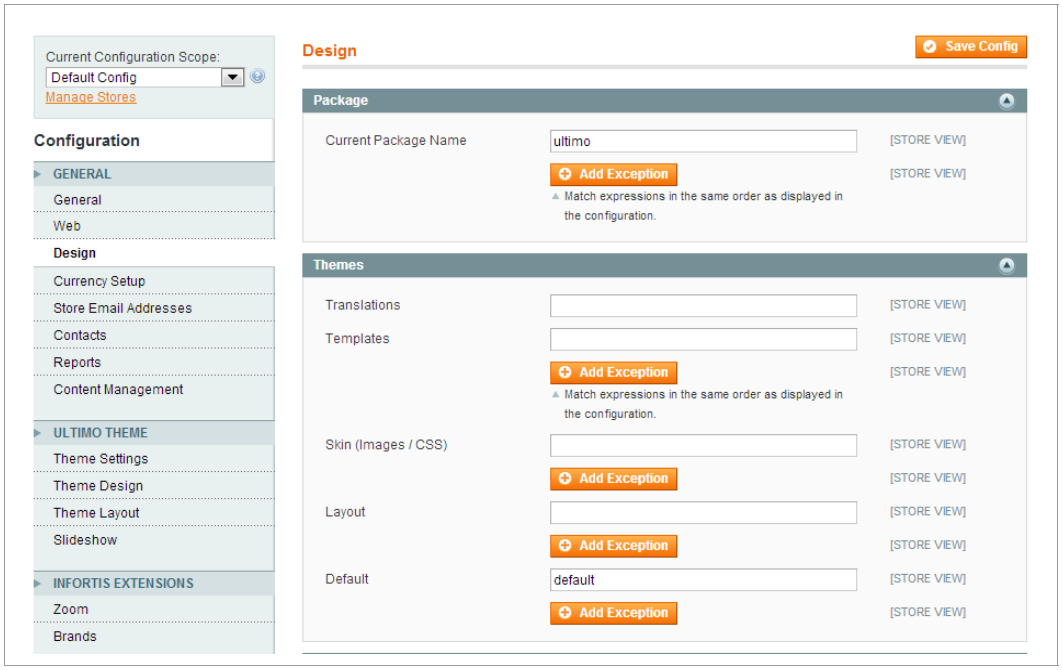
14. At this point, you need to **disable maintenance mode** in your Magento (if you enabled it earlier). To do this you need to remove file `maintenance.flag` which

you created inside the root directory of Magento. After that, you will be able to log in to the admin panel again.

> **Note:**
>
> if you get **404 not found** message when trying to open theme configuration pages in the admin panel after installation, you need to **log out** and log in again to refresh the access control system.

15. **Enable the theme**. Log in to the admin panel again. Please note that if you didn't log out before uploading the theme files (like it was described earlier in this chapter), you need to click the **Log Out** link now in order to refresh the access control system. After that you can log in again.

    Go to `System > Configuration > Design` section.



In the **Package** tab in the **Current Package Name** field enter the name of the design package: `ultimo`. Next, go to the **Themes** tab and in the **Default** field enter the name of the main theme in the package: `default`. Click **Save Config** button.

> **Important:**
>
> do not enable the theme in `System > Design`. It should only be enabled in `System > Configuration > Design`.

If you have more than one store view in your Magento (by default there's only one) and you want to enable the theme for the entire site, make sure that you select **Default Config** under **Current Configuration Scope** in the top left corner of this section. Otherwise, set the scope according to your needs.

16. Flush Magento cache (refer to **4.1.1 Flush cache**) and any other cache that you have in your Magento installation.

17. Go to the next chapter to import sample blocks and pages.

3.2.3 Import sample blocks and pages

18. You can import sample static blocks and pages provided with the theme. Go to `System > Configuration > Ultimo > Theme Settings > Theme Installation` to import static blocks and pages. Refer to **7.12 Data Import** for more details about the import.

Now you can open up your web browser to see the new look of the store. Clear your web browser's cache and cookies for your store domain (this will eliminate potential problems with unrefreshed cache).

> **Note:**
>
> if you get **404 not found** message when trying to open theme configuration pages in the admin panel after installation, you need to **log out** and log in again to refresh the access control system.

All the theme settings in the admin panel can be found under these two tabs:

- `System > Configuration > Ultimo`
- `System > Configuration > Infortis Extensions`

In case of any problems after installation, please refer to the next chapter.

3.3 Problems after installation

> **Important:**
>
> **disable Magento cache** before you start configuring your store. Otherwise you will need to flush the cache in order to see any changes you have made in the configuration.

If you encounter any problems after installation: flush Magento cache, log out from the admin panel, and log in again. Check again the installation steps and make sure you didn't omit any steps.

What can cause problems:

a) Magento cache (or external cache) was not flushed after installation of the theme.

b) CSS files which are responsible for the visual appearance of the theme were not generated automatically. Go to `System > Configuration > Ultimo > Theme Design` and click **Save Config** button. Then go to `System > Configuration > Ultimo > Theme Layout` and click **Save Config** button again. Then, flush Magento cache and refresh your browser's cache.

c) You can see all sorts of error messages if the **Compilation** mode was not disabled before installation.

d) Some of the theme files were not uploaded or are broken. Re-upload original theme files to your server (overwrite existing files) and flush the cache. Note that files can get broken if you send them in large numbers via FTP without zipping. So it is highly recommended to upload zipped files.

e) File permissions issue – refer to **14.1.4 File permissions** for more details.

f) Theme wasn't enabled properly. Theme should only be enabled in `System > Configuration > Design` (the same as it was described in the installation steps). Refer to **4.3 How to enable the theme** for more details.

g) You have some third-party extensions which change/break theme's default behavior or cause some conflicts. Disable all extensions and flush the cache, then enable extensions one by one to check which one is causing problems. Refer to **2.3 Third-party extensions** for more details about third-party extensions.

h) You edited theme files and accidentally removed or changed something in the files. Re-upload original theme files to your server (overwrite existing files) and flush the cache.

i) You have customized the theme by creating custom sub-theme and your modifications override or break the default behavior of the theme.

j) You have some elements that left from your previous theme (e.g. in Magento's "base" theme in `app/design/frontend/base/`) and that breaks the new theme.

Also check your log files in `var/log/` (log settings should be enabled in *`System > Config > Developer`*) for more information about possible errors.

Please refer to the following chapter: **14. Troubleshooting** for more information about the most common problems.

4. Magento configuration

The following chapter describes some of the basic Magento settings and features.

Please note that as authors of this theme we provide support only for the issues related strictly to the theme. Support for Magento configuration/installation/maintenance etc. is beyond the scope of our support. This chapter should be only treated as a starting point, more information about Magento can be found in Magento documentation: **2.1.2 Magento help**

4.1 Caching system

When developing your Magento store, you want to see the changes you have implemented. In order to be able to see changes immediately, you should completely **disable the cache**. You can enable it after you finish configuring your store.

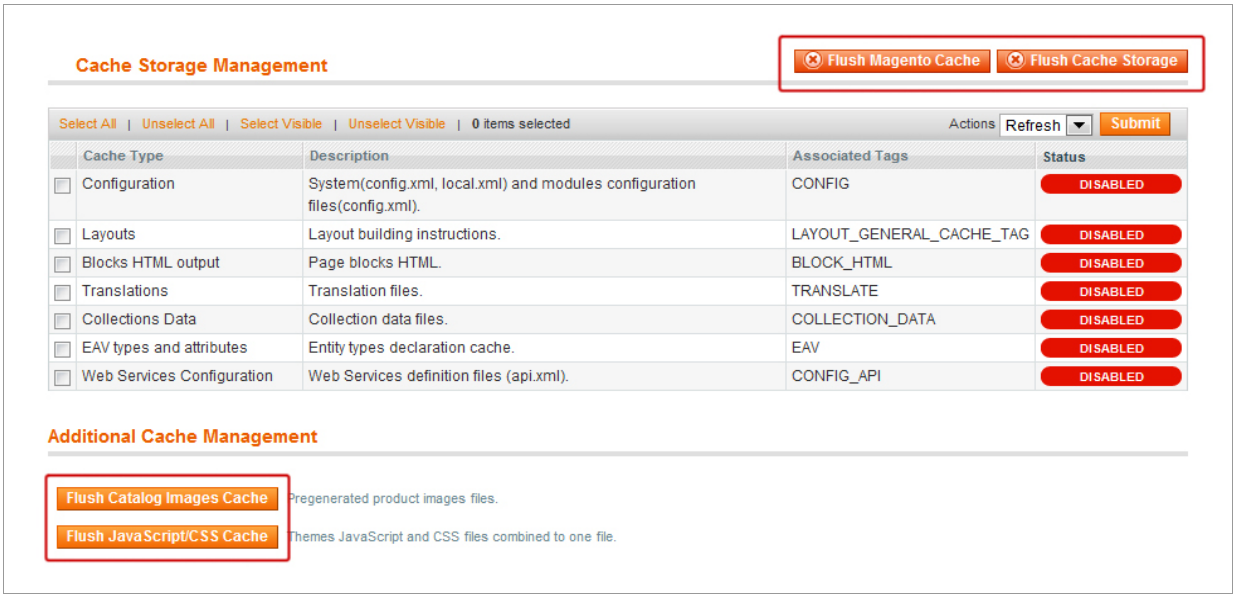
If the cache is enabled, always **refresh the cache** after making any changes in Magento/theme/extension configuration.

To access the cache management screen, go to `System > Cache Management`.

4.1.1 Flush cache

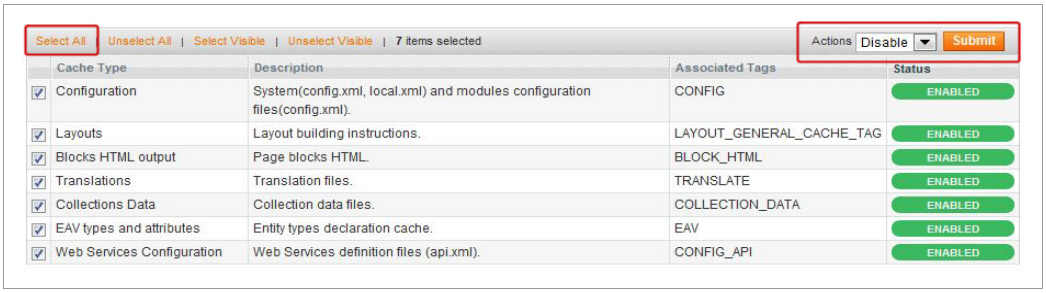
Go to `System > Cache Management`. Click the following buttons to completely clear the store cache:

1. Flush Magento Cache
2. Flush Cache Storage
3. Flush Catalog Images Cache
4. Flush JavaScript/CSS Cache

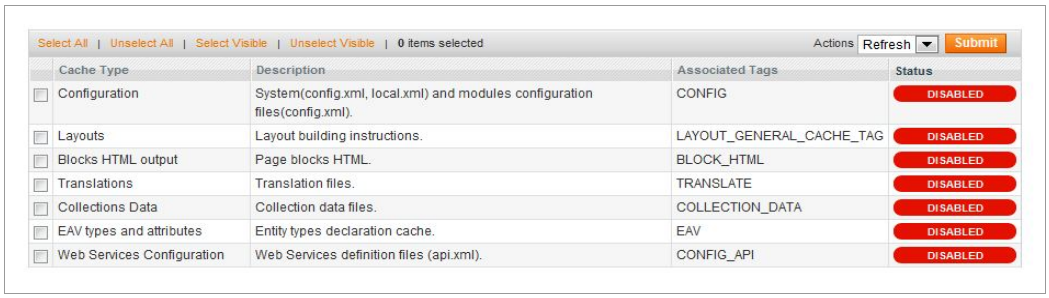


4.1.2 Disable cache

Go to `System > Cache Management`. Click **Select All** link, select **Disable** in **Actions** field and click **Submit** button:

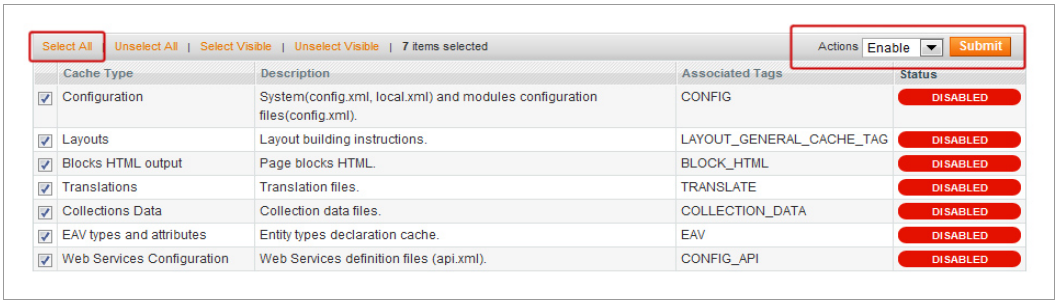

Select All | Unselect All | Select Visible | Unselect Visible | 7 items selected
Actions Disable Submit
Cache Type | Description | Associated Tags | Status
Configuration | System(config.xml, local.xml) and modules configuration files(config.xml). | CONFIG | ENABLED
Layouts | Layout building instructions. | LAYOUT_GENERAL_CACHE_TAG | ENABLED
Blocks HTML output | Page blocks HTML. | BLOCK_HTML | ENABLED
Translations | Translation files. | TRANSLATE | ENABLED
Collections Data | Collection data files. | COLLECTION_DATA | ENABLED
EAV types and attributes | Entity types declaration cache. | EAV | ENABLED
Web Services Configuration | Web Services definition files (api.xml). | CONFIG_API | ENABLED

After cache is disabled cache management screen might look like this:


Select All | Unselect All | Select Visible | Unselect Visible | 0 items selected
Actions Refresh Submit
Cache Type | Description | Associated Tags | Status
Configuration | System(config.xml, local.xml) and modules configuration files(config.xml). | CONFIG | DISABLED
Layouts | Layout building instructions. | LAYOUT_GENERAL_CACHE_TAG | DISABLED
Blocks HTML output | Page blocks HTML. | BLOCK_HTML | DISABLED
Translations | Translation files. | TRANSLATE | DISABLED
Collections Data | Collection data files. | COLLECTION_DATA | DISABLED
EAV types and attributes | Entity types declaration cache. | EAV | DISABLED
Web Services Configuration | Web Services definition files (api.xml). | CONFIG_API | DISABLED

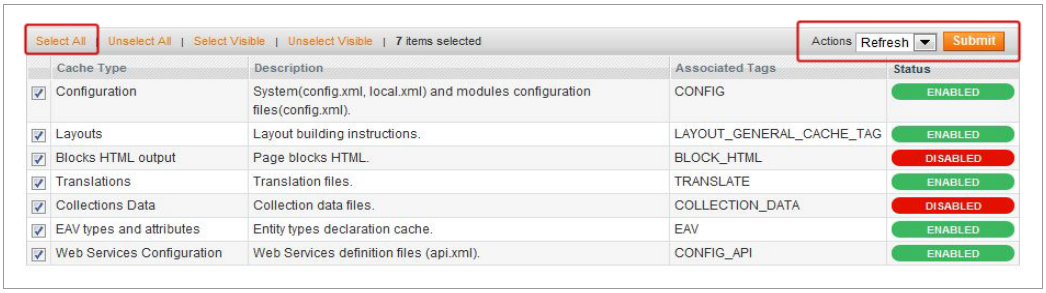
4.1.3 Enable cache

When the store is configured, you can enable caching system again. Go to `System > Cache Management`. Click **Select All** link, select **Enable** in **Actions** field and click **Submit** button:


Select All | Unselect All | Select Visible | Unselect Visible | 7 items selected
Actions Enable Submit
Cache Type | Description | Associated Tags | Status
Configuration | System(config.xml, local.xml) and modules configuration files(config.xml). | CONFIG | DISABLED
Layouts | Layout building instructions. | LAYOUT_GENERAL_CACHE_TAG | DISABLED
Blocks HTML output | Page blocks HTML. | BLOCK_HTML | DISABLED
Translations | Translation files. | TRANSLATE | DISABLED
Collections Data | Collection data files. | COLLECTION_DATA | DISABLED
EAV types and attributes | Entity types declaration cache. | EAV | DISABLED
Web Services Configuration | Web Services definition files (api.xml). | CONFIG_API | DISABLED

4.1.4 Refresh cache

If the cache is enabled, after each configuration change, extension installation, upgrade or uninstallation always refresh the cache. To do this, go to `System > Cache Management`. Click **Select All** link, select **Refresh** in **Actions** field and click **Submit** button:



4.2 Index management

Magento indexes most of its data in order to access it faster. If you make changes to your store such as editing products, adding categories, changing urls, etc. you will need to reindex the data so that the changes can show up on your frontend.

To access the index management screen, go to `System > Index Management`. To rebuild your indexes, select all, select **Reindex Data** and click **Submit** button.

> **Important:**
>
> you need to reindex after making any changes in Magento catalog: categories, products, attributes etc.

4.3 How to enable the theme

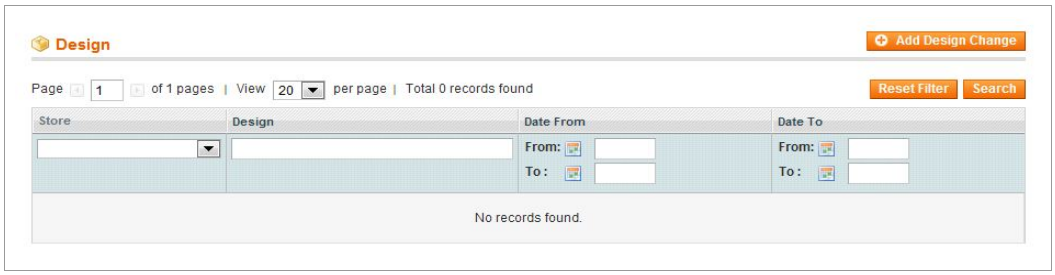
In Magento you can enable the theme on many levels. You can set one theme for the entire store, but you can also set different themes for different parts of the store (even for selected categories, products or CMS pages).

If you want all your store pages to look the same (to use the same theme), **your theme should only be enabled** in `System > Configuration > Design` (the same as it was described in the installation steps **3. Installation**). You need to remove all theme overrides from categories, products and CMS pages.

There are many places where a theme can be enabled. If you have just installed a fresh copy of Magento, those places will be empty. Otherwise, please take a look at these sections of the admin panel:

- `System > Design`

  Ultimo should not be enabled through this section, delete all existing entries related to Ultimo. By default this section looks like this:



- These fields should be left empty, do not select any theme here:

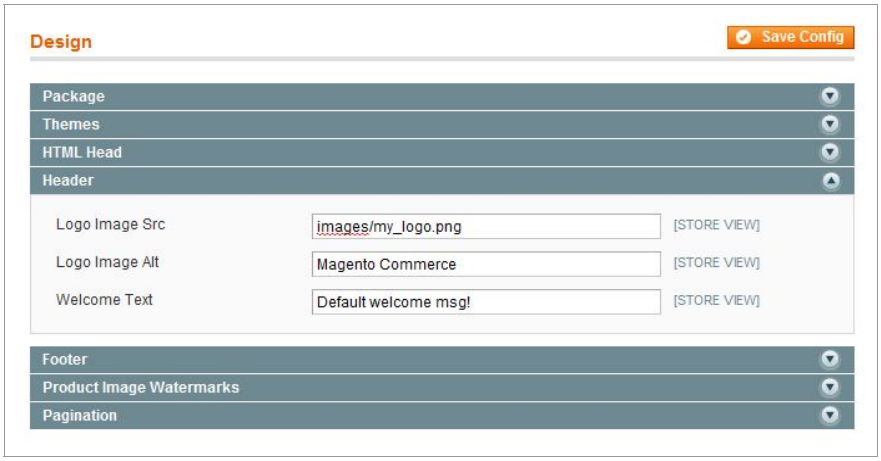
  - `Catalog > Manage Categories > [category] > Custom Design > Custom Design`
  - `Catalog > Manage Products > [product] > Design > Custom Design`
  - `CMS > Pages > [page] > Design > Custom Theme`

Leave `--Please Select--` in these fields, like on the following screenshot:

Remember to **flush the cache** after you change design settings.

4.4 Logo

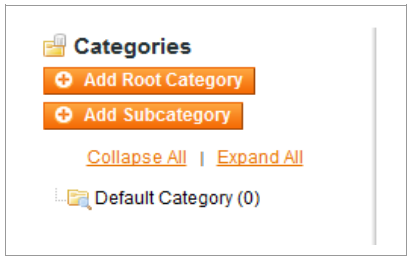
Upload your logo to `skin/frontend/ultimo/default/images/` directory. By default Magento displays logo from the file named **logo.gif**. If you uploaded file with different name or different file format (e.g. *.jpg* or *.png*), go to *System > Configuration > Design > Header* section and in the **Logo Image Src** field specify a path to your logo image. The path is relative to `skin/frontend/ultimo/default/` directory.



4.5 Categories

> **Note:**
>
> More info about categories in Magento can be found in: *http://www.magentocommerce.com/wiki/welcome_to_the_magento_user_s_guide/chapter_3#creating_categories*

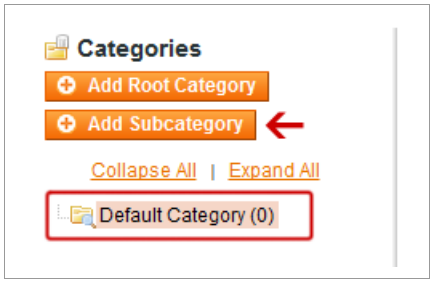
To create and edit categories in Magento, navigate to *Catalog > Manage Categories* section of the admin panel. After installation of Magento your category tree is almost empty, there is only the root category (named `Default Category`):



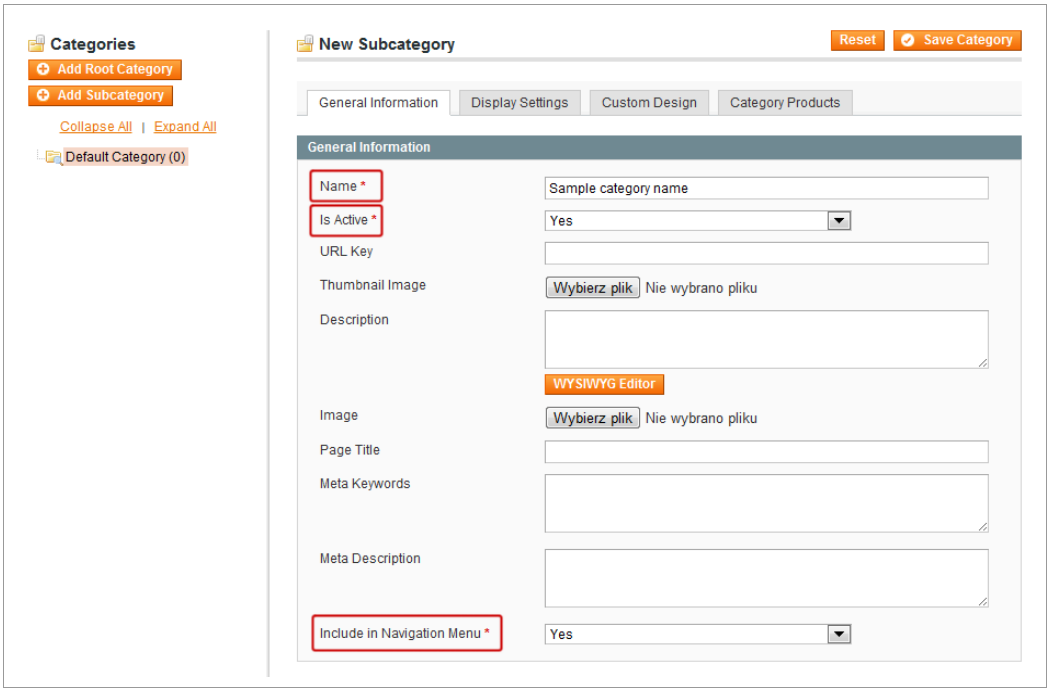
The root category is not displayed in the frontend of the store – **that's why your main menu is empty** just after installing Magento. You need to add some new top-level categories (also called subcategories because they are descendants of the root category).

> **Important:**
>
> all categories have to be **descendants** of the root category. Do not add categories outside of the root category tree.

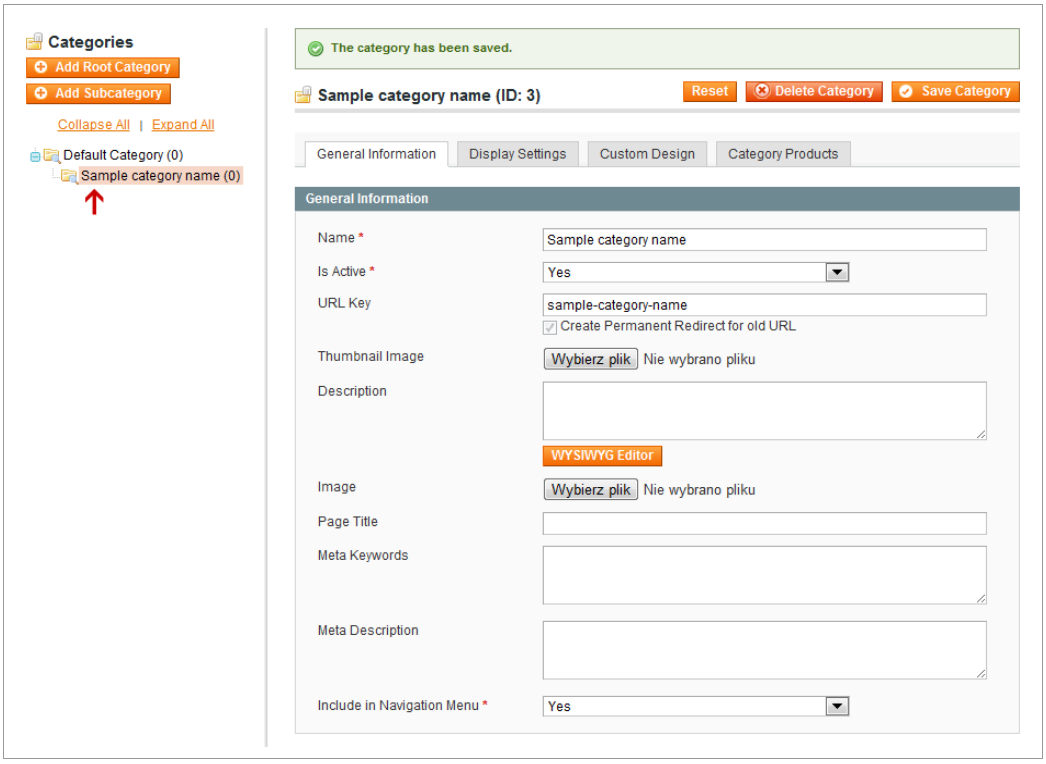
To add a new category, select the root category by clicking on it and click **Add Subcategory** button:



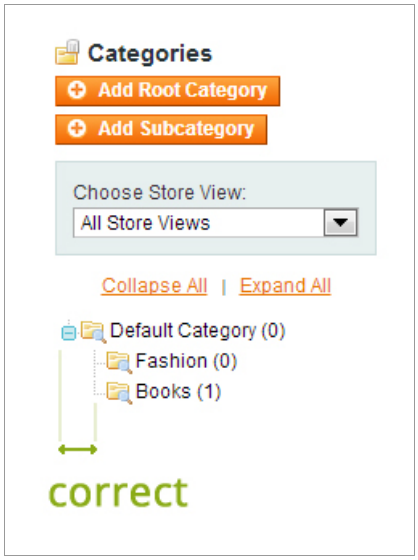
Fill in all the required fields. Especially set **Is Active** and **Include in Navigation Menu** fields to **Yes**, and click **Save Category**:

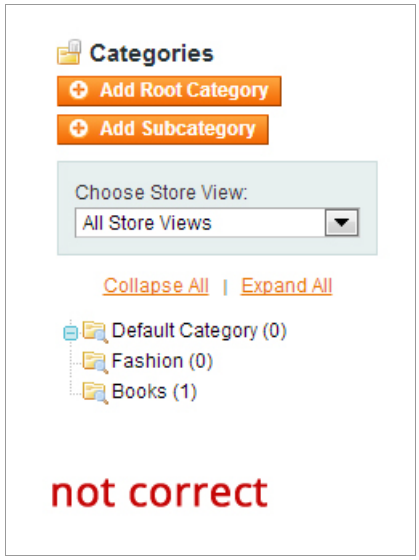


This will create a new category (which will be displayed as a top-level category in the main menu on frontend):



Note that this new category has to be a **descendant of the root category** – it has to be one level lower than the root category:





If the category is not displayed in the frontend: **reindex the data** in `System > Index Management` and **flush Magento cache**.

4.5.1 Category info

For each category you can add some basic information which will be displayed on category page (above the products). Navigate to *Catalog > Manage Categories* and select a category:

- **Description** – category description. You can enter text or HTML.
- **Image** – main category image. After image is uploaded and saved, an icon of the existing image will appear to the left of the button, and a **Delete Image** checkbox will appear to the right.

4.5.2 Landing pages

You can show your customers a landing page instead of the standard product listing page when they select the category. You can do so by enabling static block on your category page. Navigate to *Catalog > Manage Categories*, select a category, open **Display Settings** tab and select appropriate value in the **Display Mode** field. In the **CMS Block** drop-down, select which block you would like to display on category page. The list will include all existing static blocks.

4.5.3 Depth of the menu

In the main menu you can limit number of displayed category levels. To do that, go to *System > Configuration > Catalog* and specify the depth in **Maximal Depth** field.

For example, if you want to display only three levels of categories, enter `3` in **Maximal Depth** field and save the configuration. From now on, only three levels of categories will be displayed in the main menu.

4.5.4 Layered Navigation ("Shop By" filters)

Magento's Layered Navigation allows your customers to filter down products using any attribute that is set as filterable for Layered Navigation. The Layered Navigation menu will display in the left column of your category pages, and will contain all of the attributes which are filterable.

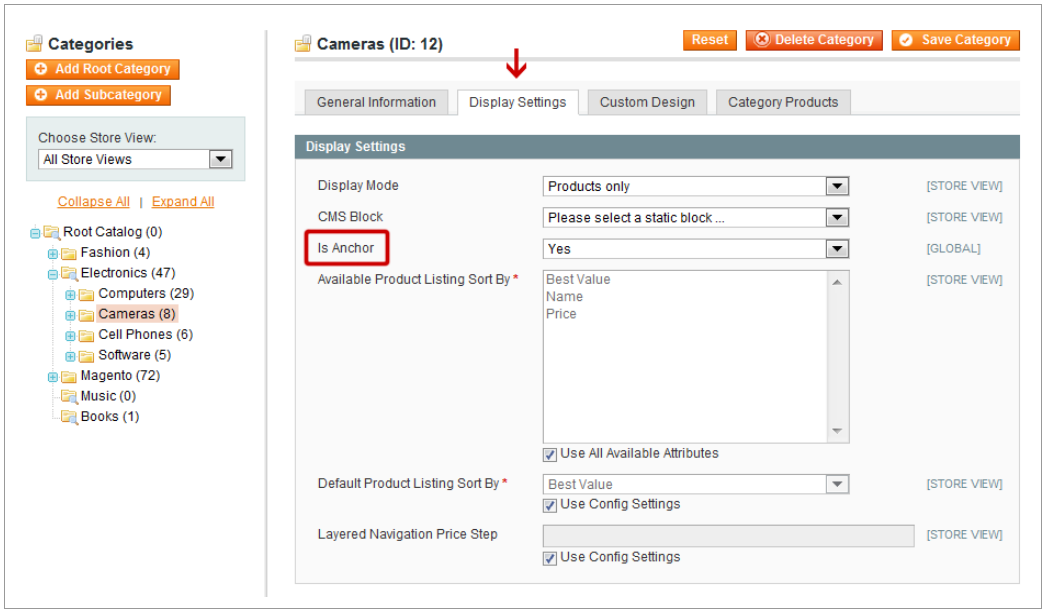
You can configure each attribute to be displayed in Layered Navigation: **Use in Layered Navigation** field in *Catalog > Attributes > Manage Attributes > [select attribute]*. Please refer to Magento documentation for more details:

- http://www.magentocommerce.com/wiki/welcome_to_the_magento_user_s_guide/chapter_3#attributes
- http://www.magentocommerce.com/wiki/welcome_to_the_magento_user_s_guide/chapter_3#layered_navigation

4.5.5 Categories inside Layered Navigation

> **Note:**
>
> This theme adds ability to display a list of categories in the category view (independent from Magento's Layered Navigation). For more details refer to **13.1.4 Sidebar Menu**.

You can display categories **inside** Magento's Layered Navigation block ("Shop by" block) at the top of the left sidebar in category view. To display selected category in the Layered Navigation, go to `Catalog > Manage Categories`, click selected category and set **Is Anchor** field (in **Display Settings** tab) to **Yes:**

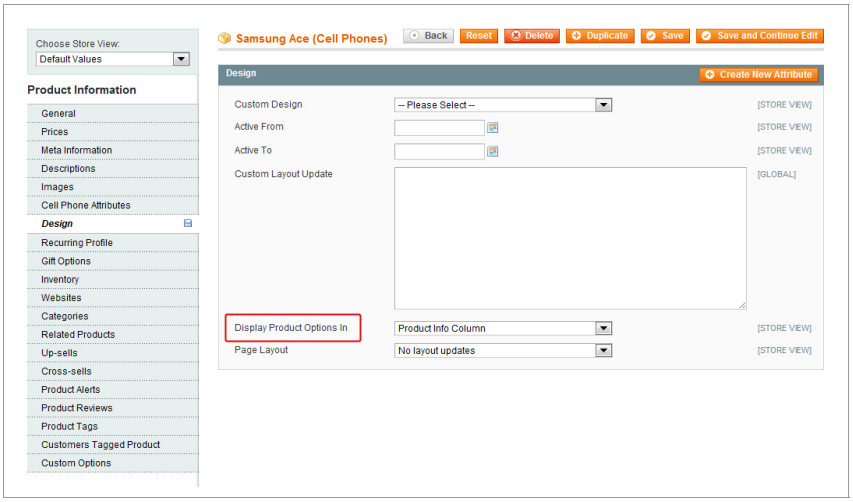


From now on this category will be displayed in the Layered Navigation. If the category is not displayed in the frontend: **reindex the data** in `System > Index Management` and **refresh Magento cache**.

4.6 Products

4.6.1 Position of: Add to Cart button and product options

For products which have options (like configurable products or bundle products) you can change the position of **options box** and **"Add to cart"** button on product page. These elements can be displayed in the main central column (the column where product name is displayed), or just below the product images (and above tabs). To set the position, select product in `Catalog > Manage Products`, open **Design** tab and select position in **Display Product Options In** field:



The default value of this field ("Block after Info Column") can't be changed anywhere in Magento admin, but instead changing the default value you can simply update that filed in all products at once: use **Update Attributes** action in the **Actions** drop-down list in `Catalog > Manage Products` section.

> **Important:**
>
> to update position of product options and "Add to cart" button on product page in many products at a time, you can use **Update Attributes** action in the **Actions** drop-down list in `Catalog > Manage Products` section.

4.6.2 "Only X left" stock indicator

There is a quantity indicator displayed on the product page. It shows the quantity of items left in stock if the stock level is equal or below the threshold specified in the admin panel. The threshold can be set in **Only X left Threshold** field in *System > Configuration > Inventory*.

For example, if you set the option to 20, all products with quantity equal or less than 20 will have the text `Only X left` on the product page.

For **configurable products** this text is clickable and shows drop-down box with quantities of all products associated with that product.

4.7 Attributes

An attribute in Magento is defined as any property of a product, such as color, price, SKU, manufacturer etc. There are two types of attributes: System Attributes (cannot be deleted and every product must contain them) and Simple Attributes (which can be created by the store admin). Attributes are grouped in Attribute Sets.

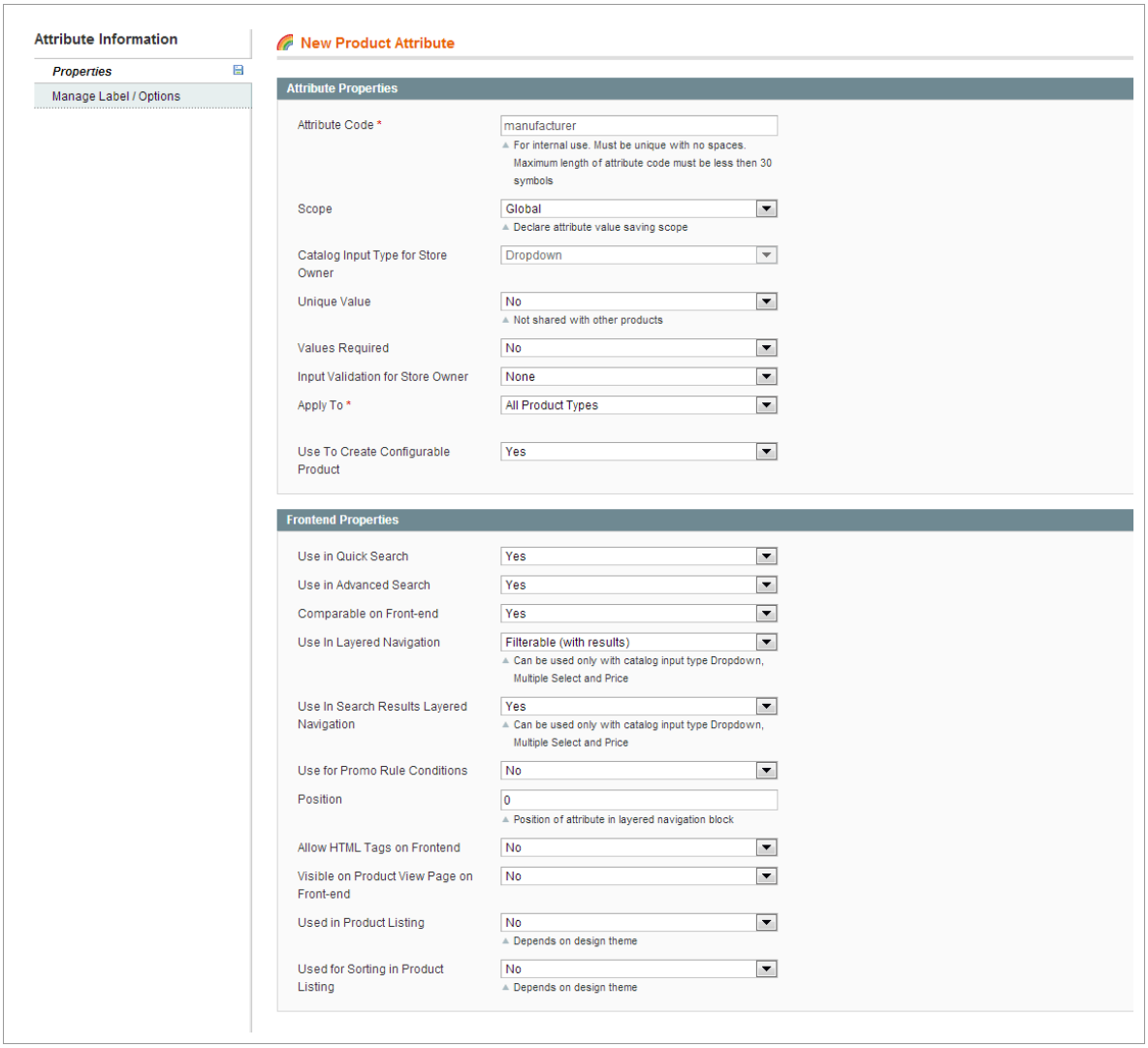
Here you can find more info about Magento attributes:
http://www.magentocommerce.com/wiki/welcome_to_the_magento_user_s_guide/chapter_3#attributes

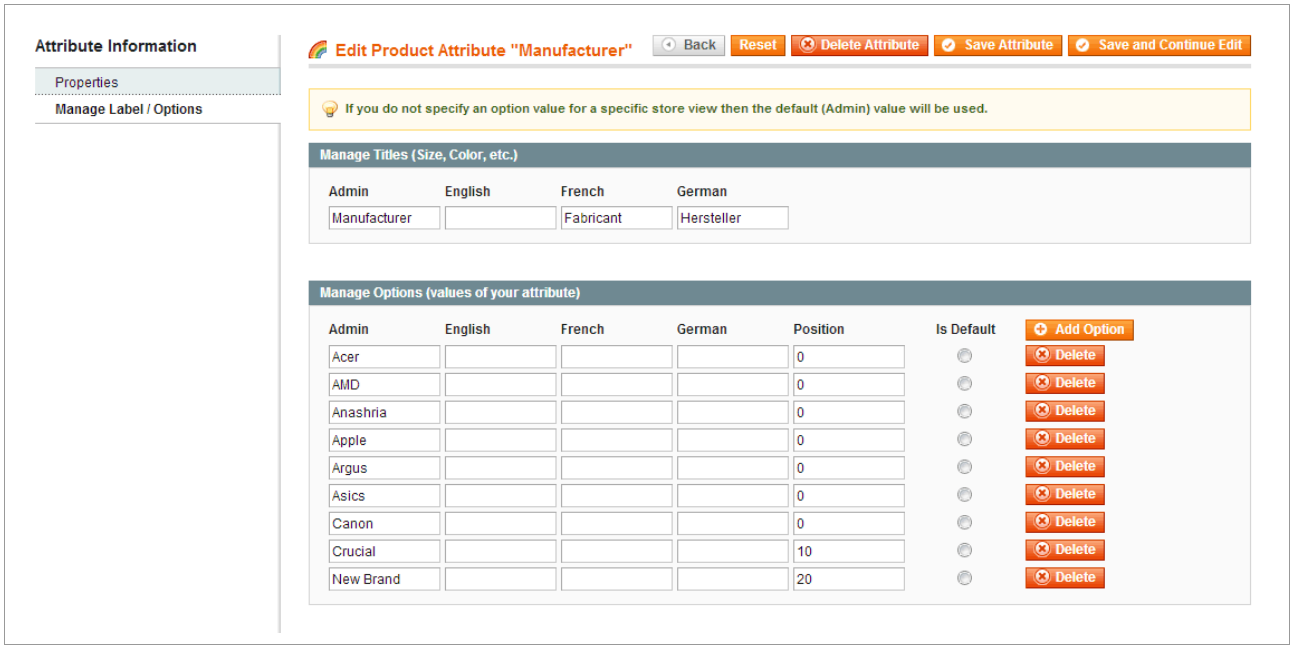
4.7.1 Attribute Management

To edit existing attributes, go to `Catalog > Attributes > Manage Attributes` and select an attribute from the list. To create a new attribute, click **Add New Attribute** button. There are many types of attributes which you can create, one of the most useful is a **drop-down** attribute which lets you define a list of values that will be available when creating new products.

> **Important:**
>
> after making any changes in the attributes and Attribute Sets, you may need to reindex the data in your Magento (**4.2 Index management**) and flush the cache (**4.1 Caching system**).

Let's see an example of a **drop-down** attribute `manufacturer` which stores a list of product brands. Fill in the form as follows:


Attribute Information
Properties
Manage Label / Options
New Product Attribute
Attribute Properties
Attribute Code *
manufacturer
For internal use. Must be unique with no spaces. Maximum length of attribute code must be less then 30 symbols
Scope
Global
Declare attribute value saving scope
Catalog Input Type for Store Owner
Dropdown
Unique Value
No
Not shared with other products
Values Required
No
Input Validation for Store Owner
None
Apply To *
All Product Types
Use To Create Configurable Product
Yes
Frontend Properties
Use in Quick Search
Yes
Use in Advanced Search
Yes
Comparable on Front-end
Yes
Use In Layered Navigation
Filterable (with results)
Can be used only with catalog input type Dropdown, Multiple Select and Price
Use In Search Results Layered Navigation
Yes
Can be used only with catalog input type Dropdown, Multiple Select and Price
Use for Promo Rule Conditions
No
Position
0
Position of attribute in layered navigation block
Allow HTML Tags on Frontend
No
Visible on Product View Page on Front-end
No
Used in Product Listing
No
Depends on design theme
Used for Sorting in Product Listing
No
Depends on design theme


Attribute Information
Properties
Manage Label / Options
Edit Product Attribute "Manufacturer"
Back
Reset
Delete Attribute
Save Attribute
Save and Continue Edit
If you do not specify an option value for a specific store view then the default (Admin) value will be used.
Manage Titles (Size, Color, etc.)
Admin
English
French
German
Manufacturer
Fabricant
Hersteller
Manage Options (values of your attribute)
Admin
English
French
German
Position
Is Default
Add Option
Acer 0 Delete
AMD 0 Delete
Anashria 0 Delete
Apple 0 Delete
Argus 0 Delete
Asics 0 Delete
Canon 0 Delete
Crucial 10 Delete
New Brand 20 Delete

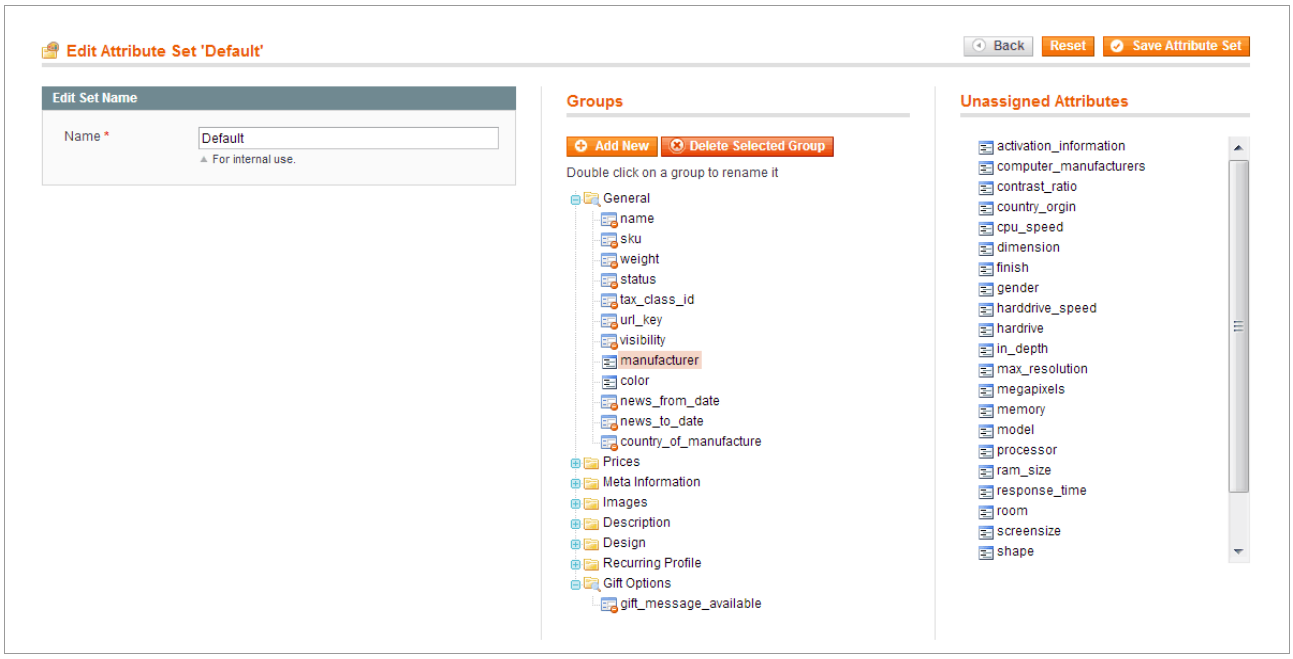
After you create the attribute, you need to add it to the Attribute Set which you use for your products. Attribute Sets are described in the next chapter.

4.7.2 Attribute Sets

An Attribute Set is a collection of attributes, created to fit certain types of products. To add a new product in Magento you need to select an Attribute Set which will describe that product. You can create many custom Attribute Sets for different types of products. For example, if you sell t-shirts and books, you could create 2 Attribute Sets: one specific to t-shirts, one specific to books. The Attribute Set for t-shirts can include attributes such as color and size (which are not needed in the Attribute Set for books).

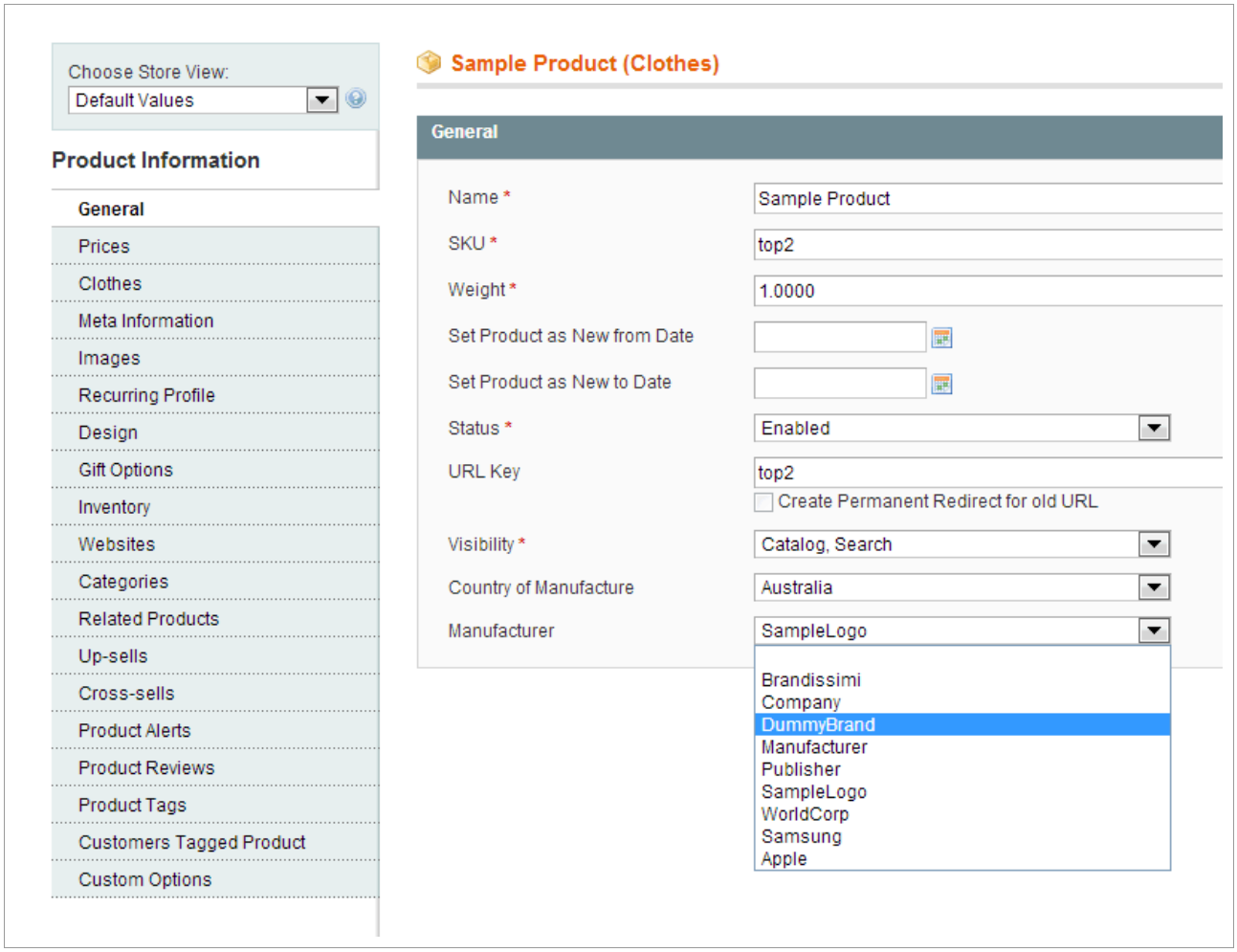
> **Important:**
>
> after making any changes in the attributes and Attribute Sets, you may need to reindex the data in your Magento (**4.2 Index management**) and flush the cache (**4.1 Caching system**).

Let's continue an example with `manufacturer` attribute. To add your attribute to an Attribute Set, go to *`Catalog > Attributes > Manage Attribute Sets`*, select an Attribute Set (default one is named `Default`). You will see **Unassigned Attributes** section with a list of available attributes. You can associate attributes to your Attribute Set by clicking and dragging an attribute into the **Groups** section. To disassociate an attribute, simply drag it back to the **Unassigned Attributes** list.



Click **Save Attribute Set** button after you finish. You can add your custom attributes to many Attribute Sets.

Then, go to *`Catalog > Manage Products`* and open a product. If the product is based on the Attribute Set which contain your `manufacturer` attribute, you will see a list of manufacturers which may look like this:



If you added `manufacturer` attribute to that Attribute Set, but you don't see the field with the list of manufacturers, you may need to reindex the data in your Magento and flush the cache.

4.8 Content Management System (CMS)

4.8.1 Add blocks to selected pages

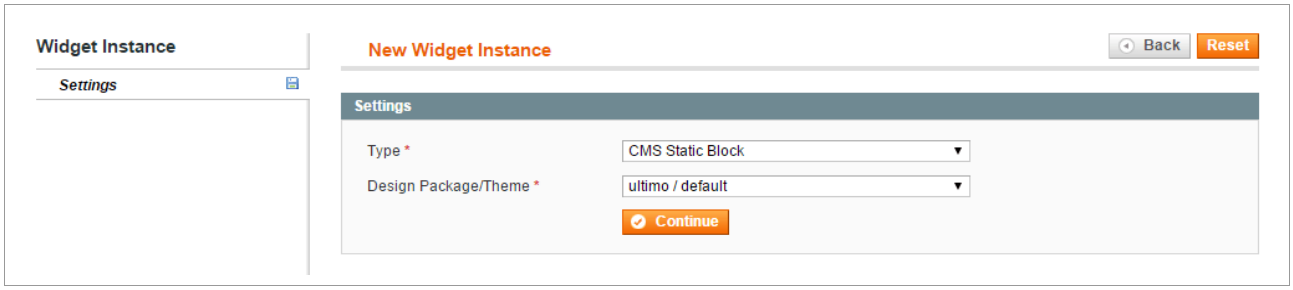
Magento Widgets (located under ...) allow users to easily add dynamic content to selected pages. This allows for greater control and flexibility in creating informational and marketing content through administrator tools.

Example:

Below you can see an example in which we will **add a simple block of content to product page**.

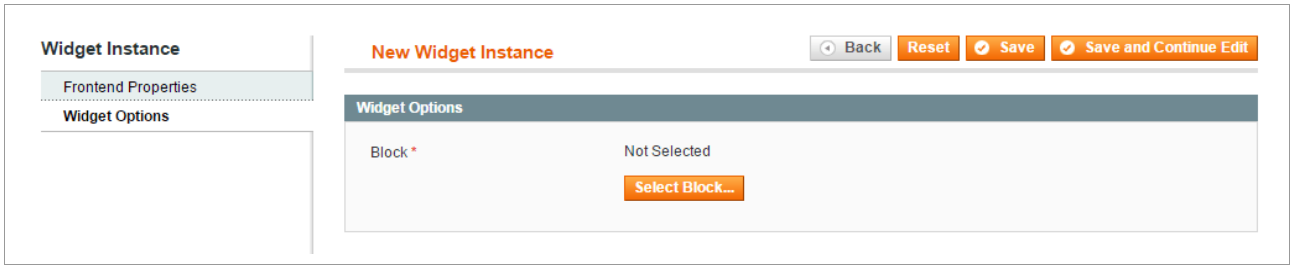
To add custom content to selected place, first create a static block. Static blocks in Magento are simple portions of content that can be displayed throughout the site. For more info about creating a static block, refer to next chapter: **4.8.2 Static blocks**.

When your static block with custom content is ready, go to `CMS > Widgets` and add a new widget instance. Widget will add your static block to a selected place on the frontend of your store. Click the **Add New Widget Instance** button. On the next page, you will see the following options:

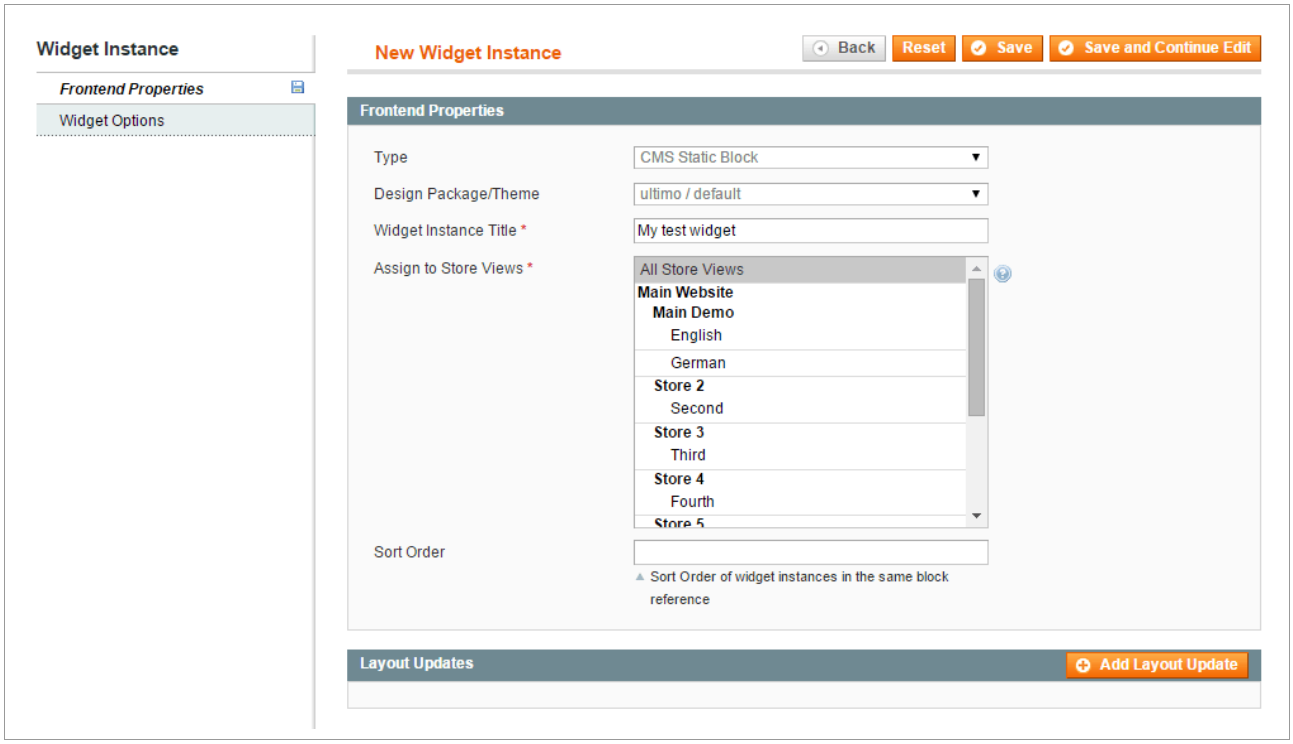


1. Choose the widget type: `CMS Static Block`.
2. Then you need to select the design in which the widget will be displayed. Select: `ultimo/default`.
3. Then, click **Continue** button.
4. On the next page, goto the **Widget Options** tab and click the **Select Block...** button to choose the static block. You will see the list of all your static block. Find and click

the one which you want to display on the frontend:



5. Next, go to the **Frontend Properties** tab.

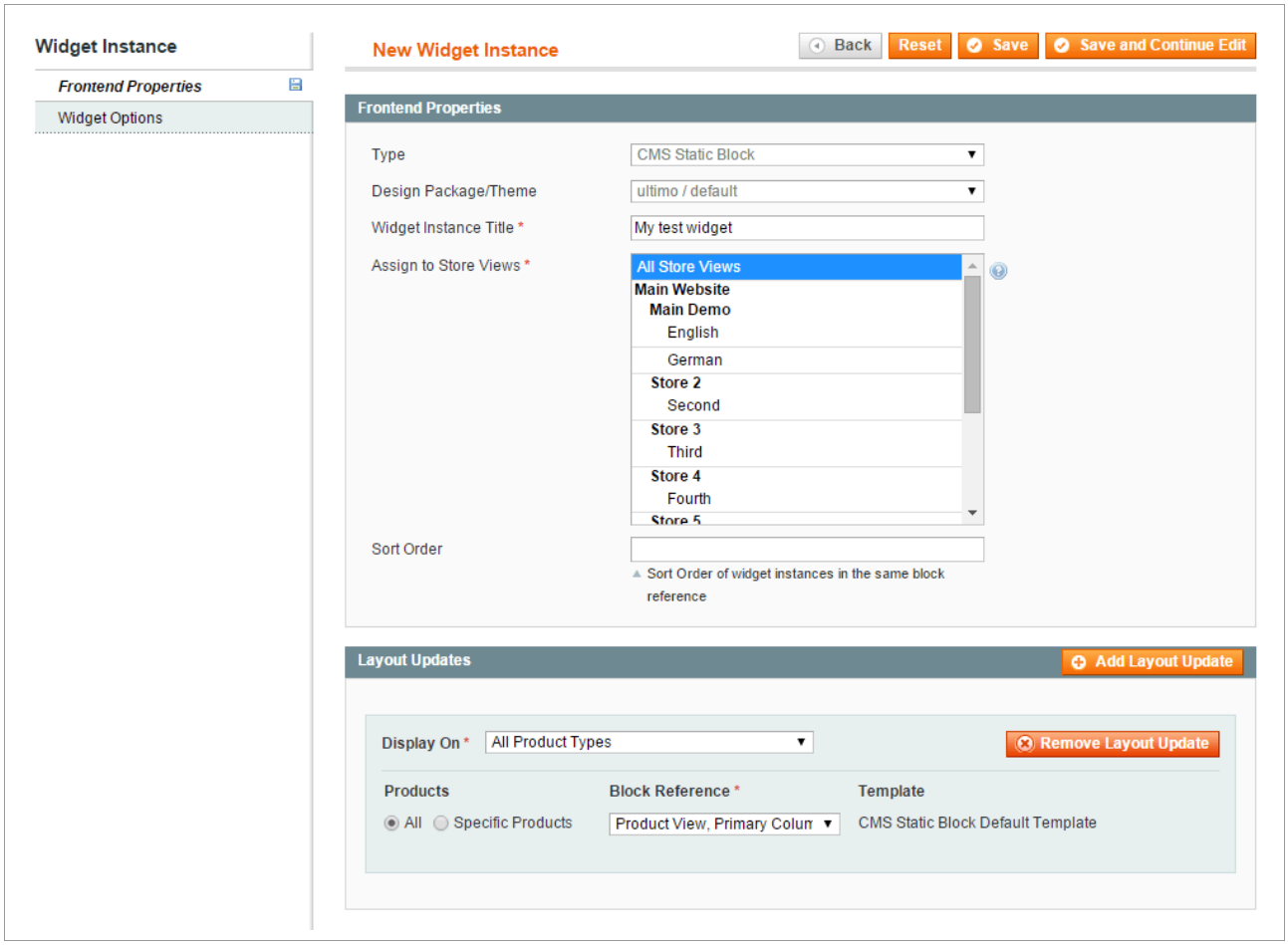


6. In the **Widget Instance Title** field give the widget a title – you can choose any name you like (try to make it informative).

7. If you have a multiple store views, in the **Assign to Store Views** field select in which store views you want to display the widget.

8. The sort order is optional, it will be used to sort widgets if more than one widget was assigned to the same container on the frontend.

9. Click on the **Add Layout Update** button – new options will be displayed inside the **Layout Updates** section:



10. In the **Display On** drop-down select a general type of pages on which the widget will be displayed. Select `All product Types`.

11. Again, new options will be displayed. You will see the contextual options for your selection. In this case, you can select for which products you want to add your widget: `All` or `Specific Products`. Select `All`.

12. In the **Block Reference** field select where exactly on the page you want to display your custom static block. Select one of the product page containers, for example: `Product View, Primary Column, Container 1`.

    List of available containers on product page can be found in this chapter **6.2.4 Product page**.

    You will notice some additional containers, the list may differ in your specific case – additional containers may have been added by any additional extensions you have installed in your Magento.

13. Finally, save the widget and then flush Magento cache to see the effect on the frontend.

4.8.2 Static blocks

Static blocks in Magento are simple portions of content that can be displayed throughout the site. To manage static blocks, go to `CMS > Static Blocks` section in the admin panel.

To create a static block, go to `CMS > Static Blocks`, click **Add New Block** button and follow these steps:

1. In **Identifier** field enter an identifier of one of the static blocks. Make sure the identifier is lower-case and separated by underscores to follow Magento's standards.

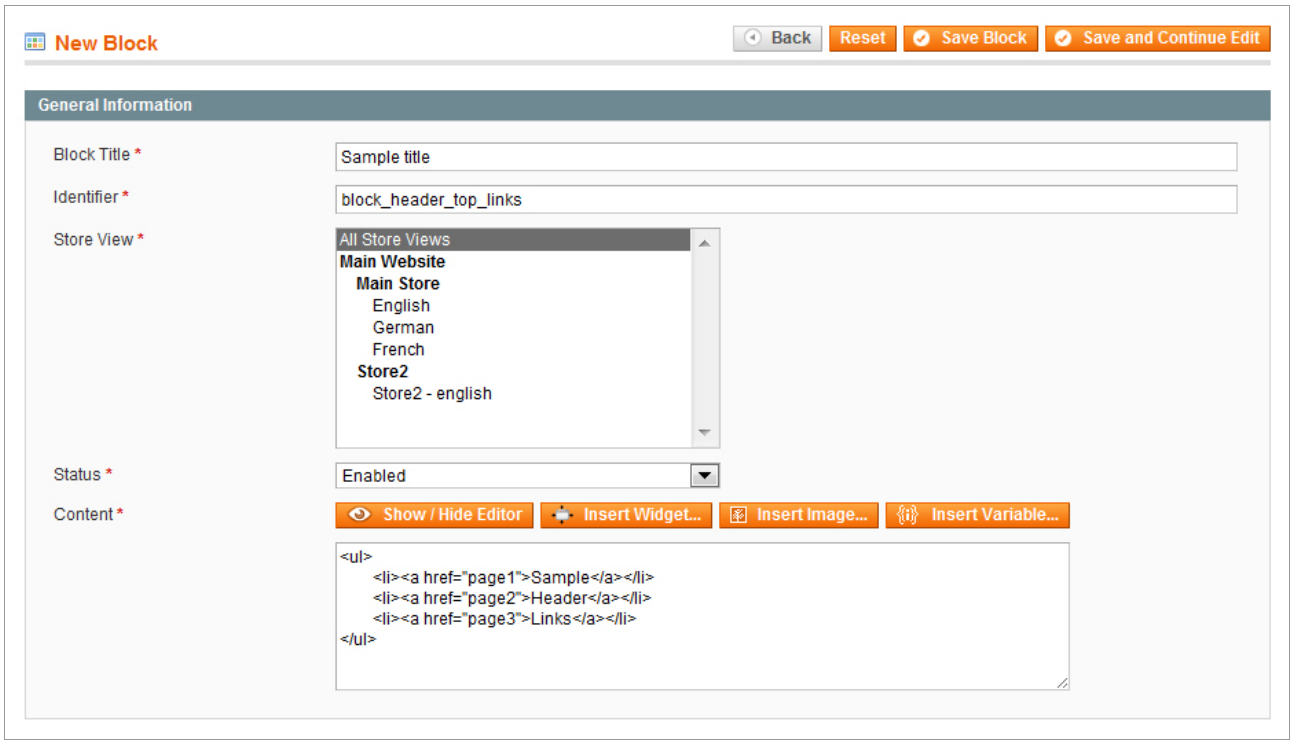
2. Enter the title in **Block Title**.

3. Select the **Store View** to which this block will apply.

4. Select `Enabled` in the **Status** field. `Disabled` means that the block is not displayed in the frontend.

5. Insert your **Content** (it can be text or HTML) and click **Save Block**.

Here's an example:



For more details on creating static blocks please refer to Magento user guide: *static blocks*

4.8.3 CMS markup tags

There are some special markup tags – a bits of text surrounded by double curly braces – which have a special meaning in Magento. You can use those tags either in static blocks or CMS pages.

For example, this tag:

```
{{store url=''}}
```

will be replaced with the store's base URL.

It might be useful when you want to display a link on one of your pages. You can use this tag to dynamically build all of your links (you don't need to hard-code any links). Thanks to that you will not need to rebuild your links after you move your store to a new domain.

For example, if your store domain is `www.example.com`, the following tag placed in the page content:

```
{{store url='path/to/page/about-us'}}
```

will be replaced with this URL:

```
http://www.example.com/path/to/page/about-us/
```

For more information, please refer to this article, where you can find a comprehensive list of available tags:
http://www.magentocommerce.com/wiki/3_-_store_setup_and_management/cms/markup_tags

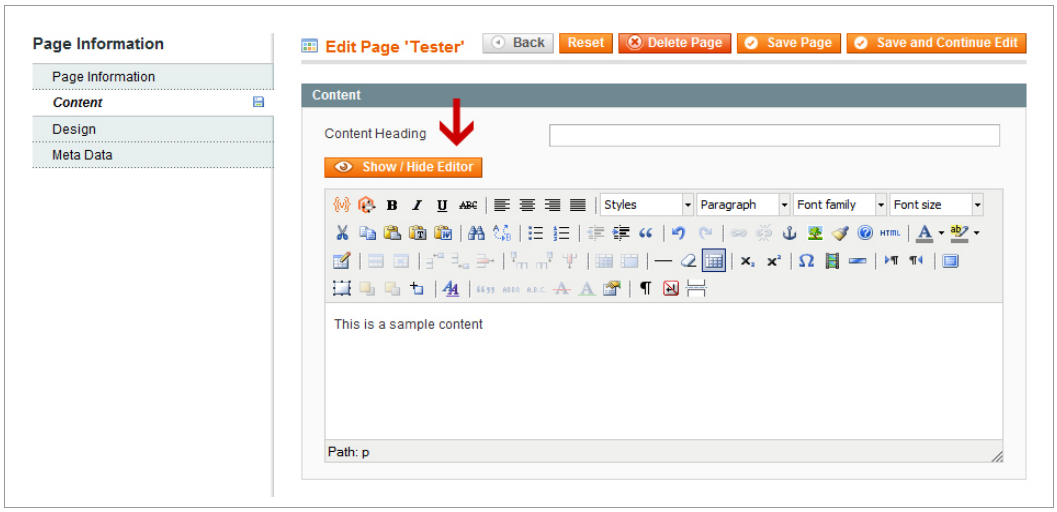
4.8.4 WYSIWYG editor

WYSIWYG (*What You See Is What You Get*) editor is a convenient way to add content to pages and static blocks in Magento.

> **Important:**
>
> WYSIWYG should only be used to edit text, **never use it to edit HTML**.
>
> Always turn off WYSIWYG editor when you want to edit HTML or Magento's CMS tags in the page/block content. Otherwise WYSIWYG editor can break your content.

But the best way to use WYSIWYG is to **turn it on only when it is needed**. Go to `System > Configuration > Content Management` and set it as **Disabled by Default.**

After that, if needed, WYSIWYG can always be turned on with **Show/Hide Editor** button above the page/block content field:



4.9 Translation / Localization

4.9.1 Interface translation

This theme introduces a few additional interface character strings. For example “Special Price” label is replaced with shorter “Now only” label. Strings are located in `app/design/frontend/ultimo/default/locale/en_US/translate.csv`. This file can be used to translate the interface into other languages.

Example:

To translate the interface into Spanish follow these instructions:

- Create a new folder for your translation. For Spanish language it will be `app/design/frontend/ultimo/default/locale/es_ES`.
- Copy `translate.csv` from `app/design/frontend/ultimo/default/locale/en_US` and paste it into created folder `app/design/frontend/ultimo/default/locale/es_ES`.
- Open `app/design/frontend/ultimo/default/locale/es_ES/translate.csv` in Open-office Calc (or other text editor like Notepad++, PSPad).

> **Note:**
>
> The editor should be capable to save file in UTF-8 encoding. Do not use Excel, it can break file structure. Read more about useful tools:
>
> *http://www.magentocommerce.com/wiki/groups/166/useful_tools_to_work_with_translations*

- If you open `translate.csv` in Open-office Calc it might look like this:

Special Price:	Now only:
My Wishlist	Wishlist
My Wishlist (%d item)	Wishlist (%d)
My Wishlist (%d items)	Wishlist (%d)

My Account	Account
…	

The first column contains the original character strings. It should be **left intact**. In the second column you can place your translation of each string.

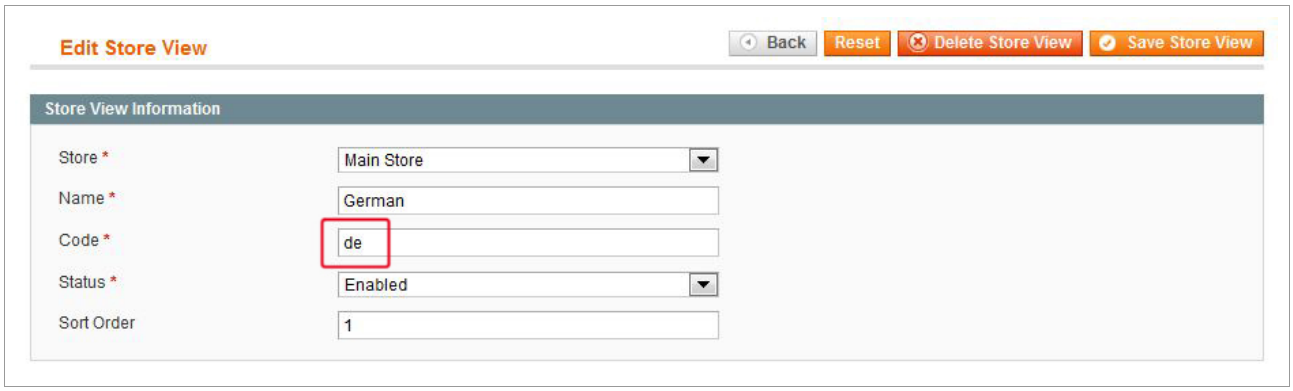
- Some strings contains `%d` or `%s` entries. These entries should be **left intact** in the translated strings.

4.9.2 Language flag

If you have more than one store view in your store, the store view switcher will be displayed at the top of the page. You can enable different language for each store view.

For each available language a flag is displayed in the store view switcher (we can also call it “language switcher”). Flag images (16x12 pixels, PNG format) should be uploaded to `skin/frontend/ultimo/default/images/flags` folder. Image names should be the same as the store view codes. E.g. if you have a store view with the code `de`, you will need to upload a flag image `de.png` to `skin/frontend/ultimo/default/images/flags` folder.

To check what is the store view code, go to *`System > Manage Stores`* and click on the store view name. You can find the code in the **Code** field:



Remember that if you change the store view code, you will also need to change the name of the corresponding flag image.

5. Customization

> **Important:**
>
> please note that Magento theme customization is far beyond the scope of this user guide. This chapter is only for informational purposes, you should treat it as a starting point. Before you start to customize the theme you need to be sure that you know what you're doing.

Magento is based on a theming concept referred to as parent/child theming – changes in the theme can be implemented without modifying the original. Magento's “base” theme serves as a parent theme for any custom theme.

The correct way to customize the theme in Magento is to create your own sub-theme of the default theme in the design package. **You should never edit original files of the design package**. Files that need to be modified can be copied from the default theme to your custom sub-theme inside the design package. But don't copy all files, only the files which you're going to modify.

This way you can override any file of the theme. And you can be sure that when something goes wrong you can simply delete your custom sub-theme without breaking the original theme files.

In this chapter you can find the information that will help you to modify the theme.

5.1 **How to modify the theme** **53**
- 5.1.1 Customize images 53
- 5.1.2 Override default CSS styles 54
- 5.1.3 Modify template files 55
- 5.1.4 Create custom sub-theme 56

5.2 **Magento theme customization** **57**
- 5.2.1 Design packages 57
- 5.2.2 Theme structure 58
- 5.2.3 Fallback mechanism 59
- 5.2.4 Examples of custom sub-theme 61
- 5.2.5 Summary 64

5.1 How to modify the theme

5.1.1 Customize images

Custom content images

All images which are used in this theme (such as page backgrounds, patterns, icons, banners inside blocks etc.) are stored in `media/wysiwyg/infortis/ultimo/` directory:

- `_patterns` – patterns which can be applied for the main sections of the page in *System > Configuration > Ultimo > Theme Design*.

  To add custom pattern, rename one of the files which name starts with `default.`, for example:

  `default.custom1.png → custom1.png`

  (by removing the first part of the name: `default.` – the dot also has to be removed). Then, replace that file with your custom image. Custom image will be available in the admin panel as custom pattern.

- `custom` – images which can be used as page background and other images for static blocks. You can create subdirectories here for custom images.

- `icons` – icons which can be used in static blocks and pages.

- `slideshow` – images for the home page slideshow.

- `social` – social icons in the footer.

Interface images

All interface images and icons are stored in `skin/frontend/ultimo/default/images/` directory. You can replace those files with custom files. You can also use **PSD** files included with this theme to create your own versions of images.

5.1.2 Override default CSS styles

When you want to make some non-standard design changes (for which you can't find any settings in the theme admin panel) you can do it by adding custom CSS styles.

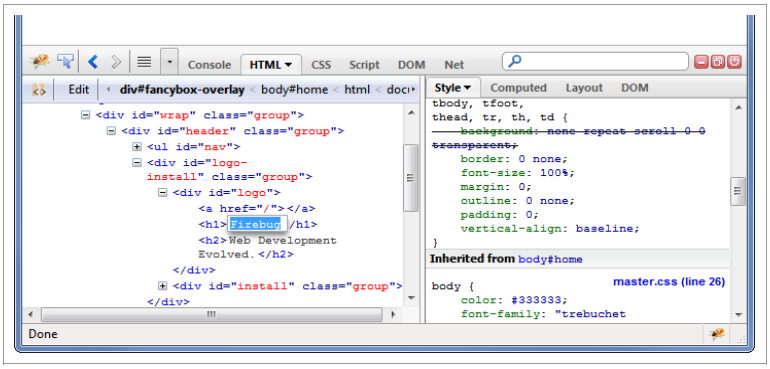
> **Note:**
>
> for simple CSS changes there's no need to create custom sub-theme.

In this theme you can enable additional CSS file `custom.css`, in which you can override and extend the default styles of the theme. All the CSS styles added to this file will not be lost after upgrading the theme in the future. Refer to **7.11 Customization Settings** to check how to enable that file.

How to find out which part of CSS to override?

How to find out which part of the theme's or Magento's CSS you need to override or extend? The best way is to use **Firebug** (an add-on for Firefox: *www.getfirebug.com*), **Chrome Developer Tools** (available in Google Chrome) or other similar tools.

See the short video tutorial about **Firebug**: *http://getfirebug.com/video/Intro2FB.mp4*



How to use CSS?

Here you can find more information about CSS:

- *https://developer.mozilla.org/en-US/learn/css*
- *http://www.w3schools.com/css/*
- *http://reference.sitepoint.com/css/syntax*

5.1.3 Modify template files

Magento consists of dozens of template files, each one is used to render a block of content in the frontend of the store. When you want to customize a theme, you can display the name of template file of every block in the frontend by enabling Magento's **Template Path Hints**.

> **Note:**
>
> to enable Magento's **Template Path Hints** go to `System > Configuration > Developer`. Select your current website under **Current Configuration Scope** in the top left corner of the admin panel, otherwise Template Path Hints settings will not be visible.
>
> See this short video tutorial for more details: *http://vimeo.com/1067069*

This way you can easily check which template files you need to edit to customize specific sections of Magento. Files that need to be modified can be copied to your custom sub-theme. Refer to **5.1.4 Create custom sub-theme** for more details about custom sub-themes.

5.1.4 Create custom sub-theme

The correct way to customize the theme in Magento is to create custom sub-theme. You should never edit the original files. Files that need to be modified can be copied to your custom sub-theme. Refer to **5.2 Magento theme customization** for more details.

> **Important:**
>
> when you're creating a sub-theme in Magneto you should only copy the files which you want to modify. **Do not copy all files.** Otherwise you will have tons of additional work in the future with any theme upgrade.

> **Important:**
>
> when you're creating a sub-theme, don't copy folder `skin/frontend/ultimo/default/css/_config/` to your sub-theme, that folder should stay in the default theme. Otherwise it will override all your theme configuration from the admin panel.

To check which template file you need to edit to customize specific section of Magento you can enable Magento's **Template Path Hints**. Refer to chapter **5.1.3 Modify template files** for more details.

If the file which you selected with **Template Path Hints** doesn't exist in Ultimo package (a theme doesn't need to override all template files from Magento's "base" theme), you can copy that file from Magento's "base" theme to your custom sub-theme inside Ultimo package.

By creating custom sub-theme:

- You can override any file.
- You can create upgrade-proof modifications.
- You can be sure that when something goes wrong you can simply delete your sub-theme without breaking the original files.

5.2 Magento theme customization

5.2.1 Design packages

Themes in Magento are grouped together into design packages (in earlier versions called “interfaces”). By default Magento Community Edition has two packages named "base" and "default".

> **Note:**
>
> do not edit files in the "base" package and do not create custom sub-themes inside the "base" package.

Every design package in Magento comes with a default theme named "default" which is the main theme in the design package. When administrator assigns a package to the store, Magento automatically looks for the theme named "default" in that package.

Design package can also contain variations of the "default" theme. This variations are often called sub-themes because they inherit all the files from the "default" theme (from the same package) and from the "base" package. Inside the sub-theme you can override any file of the "default" theme. Sub-theme can even consists of a single file – other files will be inherited from the "default" theme from the package. This is called a **fallback mechanism** because Magento always falls back to the next theme in the hierarchy to find the requested file. Fallback is described later in this chapter.

5.2.2 Theme structure

Magento themes consists of many files like HTML templates, CSS style sheets, images etc. All files are split between two directories:

- `app/design/frontend/package_name/theme_name/`

  Templates directory – contains the layout files, HTML templates and translations

- `skin/frontend/package_name/ theme_name/`

  Skin directory – contains CSS style sheets, images and theme-specific JavaScript files

where `package_name` indicates the package name and `theme_name` indicates the theme name.

Template files are organized as follows:

- `layout` – directory contains XML files which define page structure
- `template` – directory contains template files (.phtml), a mix of HTML and PHP
- `locale` – directory contains CSV files with translation strings

Skin files are organized as follows:

- `css` – directory contains CSS files
- `images` – directory contains images
- `js` – directory contains theme-specific JavaScript files

5.2.3 Fallback mechanism

Understanding fallback mechanism is very important when you want to customize the theme, so let's see a simple example.

Let's say we have a design package named "helloworld" with custom sub-theme named "new". Package files are organized as follows:

Default theme in the package:

- `app/design/frontend/helloworld/default/`
- `skin/frontend/helloworld/default/`

Your custom sub-theme in the package:

- `app/design/frontend/helloworld/new/`
- `skin/frontend/helloworld/new/`

Let's also assume that your custom sub-theme contains a CSS file called "menu.css" located in `skin/frontend/helloworld/new/css/menu.css`

If your custom theme requests "menu.css" but Magento can't find it in your custom sub-theme in `skin/frontend/helloworld/new/css/menu.css`, Magento will try to find that file in the next theme in the hierarchy. Next theme in the hierarchy is the "default" theme in "helloworld" package so Magento will search in `skin/frontend/helloworld/default/css/menu.css`.

If the file is not there, Magento will continue until it locates the file. Next theme in the hierarchy is the "default" theme in "base" package: `skin/frontend/base/default/css/menu.css`. The "default" theme in the "base" package is the **final fallback point** in the hierarchy.

Let's take a look again at the entire fallback hierarchy:

1. Look for the requested file in custom sub-theme:

   `skin/frontend/helloworld/new/css/menu.css`

2. If not found, look for the file in the "default" theme in the design package:

   `skin/frontend/helloworld/default/css/menu.css`

3. If not found, look for the file in the "default" theme in "base" design package:

   `skin/frontend/base/default/css/menu.css`

The first theme in hierarchy is the theme that you assign through the admin panel (in *`System > Configuration > Design`*). The last theme in hierarchy is the theme "default" in design package "base".

Thanks to the fallback mechanism there is no need to copy all the default theme files to your custom sub-theme when you want to make some changes in your theme. You only copy over the files that you want to edit. Other files will be inherited from the "default" theme in the package and from the "base" package.

The most important advantages of the fallback mechanism:

- selected functionality can be shared between many themes
- less code to maintain
- custom themes are update-proof: you don't edit original files, so your changes are not overwritten after theme updated

5.2.4 Examples of custom sub-theme

Let's see some examples.

Example 1

You have just installed a custom design package named "helloworld" with the default theme (named "default") inside. Let's say that you want to change the font color of the product name on product page.

Package files are organized as follows:

- `app/design/frontend/helloworld/default/`
- `skin/frontend/helloworld/default/`

The correct way to customize the "default" theme is to create your own sub-theme in which you will override **selected files** (not all files) of the "default" theme.

1. Let's call the new theme "my_world". Create the following folder inside the "helloworld" package: `skin/frontend/helloworld/my_world/`. The name of that folder is the name by which Magento will recognize your theme.

2. Create a copy of the CSS file `skin/frontend/helloworld/default/css/styles.css`

   and paste it into your sub-theme directory: `skin/frontend/helloworld/my_world/css/styles.css`

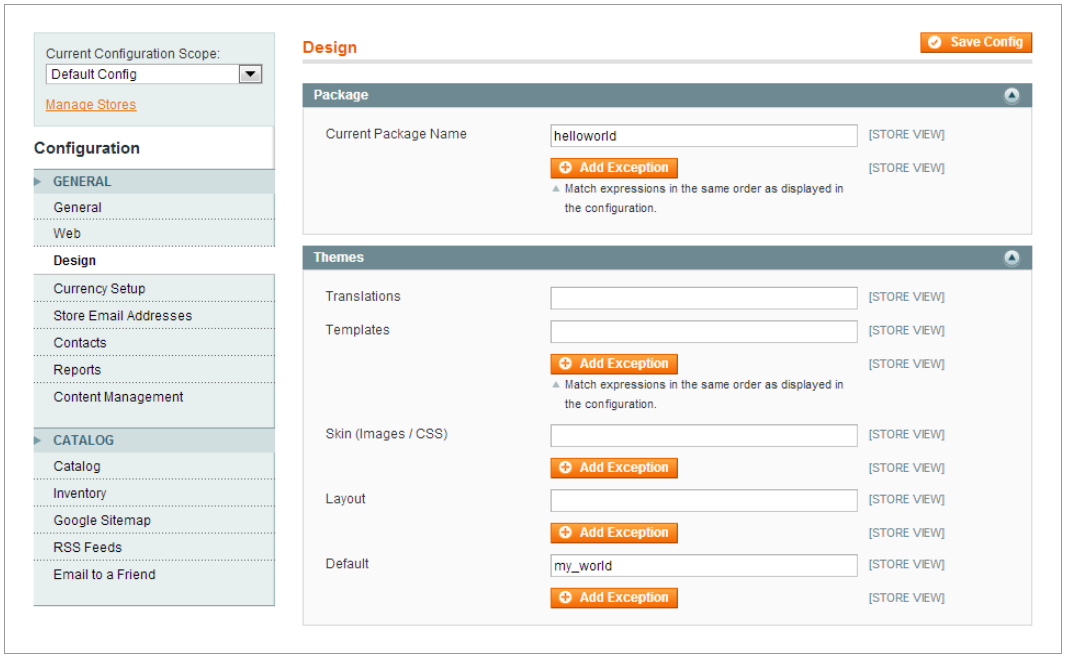
   Note that the directory structure inside the sub-theme **must replicate the directory structure of the default theme**:

   - `skin/frontend/helloworld/default/css/styles.css`
   - `skin/frontend/helloworld/my_world/css/styles.css`

3. Open `skin/frontend/helloworld/my_world/css/styles.css style sheet` and add these lines at the end of the file to override the default color of the product name:

```
.product-view .product-shop .product-name h1 { color:#f00; }
```

4. Enable the new sub-theme in the admin panel. Go to *System > Configuration > Design > Themes* and enter `my_world` (the sub-theme name) in the **default** field:



   Finally, refresh Magento cache. From now on your customization will override the default styling of the "helloworld" package.

Example 2

In the previous example we created custom sub-theme named "my_world" in the "helloworld" design package to override one of the skin files of the "default" theme. But you can also override template files located in `app/design/frontend`, not only skin files.

Continuing with the previous example, let's override the contact form template to add simple image banner above the form.

1. Create the following folder for the "my_world" template files in the "helloworld" package: `app/design/frontend/helloworld/my_world/`

   The name of the folder has to be the same as the name of the folder which you created for the skin files (`skin/frontend/helloworld/my_world/`)

2. The file which is responsible for the contact form is located here:
   `app/design/frontend/helloworld/default/template/contacts/form.phtml`

   Create a copy of that file and paste it into your sub-theme directory (remember that the directory structure inside the sub-theme **must replicate the directory structure of the default theme**):
   `app/design/frontend/helloworld/my_world/template/contacts/form.phtml`

   Note that if the `form.phtml` file doesn't exist in the "helloworld" package, you can copy it from the "base" package (keeping the directory structure intact). "base" package contains all the files that control Magento's default behavior. If the requested file is not found in your custom sub-theme, Magento will try to find that file in the "default" theme from the package. If the file is not found, Magento will call the file from the "base" package.

   Thanks to the fallback mechanism, in our example Magento will try to find the file in the following order:

   1. `app/design/frontend/helloworld/my_world/template/contacts/form.phtml`
   2. `app/design/frontend/helloworld/default/template/contacts/form.phtml`
   3. `app/design/frontend/base/default/template/contacts/form.phtml`

3. Open `app/design/frontend/helloworld/my_world/template/contacts/form.phtml` file and insert image banner before the beginning of the contact form.

4. If you didn't enable the new sub-theme in the previous example, you need to do it now. Go to *System > Configuration > Design > Themes* and enter `my_world` (the sub-theme name) in the **default** field. Finally, refresh Magento cache to display changes you have made.

5.2.5 Summary

- Themes in Magento are grouped together into design packages.
- By default Magento comes with the package named "base" which contains all the files that control Magento's default behavior.
- Every design package in Magento comes with a default theme named "default" which is the main theme of the design package.
- Design package can contain variations (sub-themes) of the "default" theme. They inherit all the files from the "default" theme (from the same package) and from the "base" package.
- The correct way to customize the theme in Magento is to create your own sub-theme. You should **never edit original design package files**. Files that need to be changed can be copied to your custom sub-theme.
- This way you can override any file of any design package. And you can be sure that when something goes wrong you can simply delete your custom sub-theme without breaking the original files.
- Magento use fallback mechanism to make custom themes easier to maintain and upgrade-proof.
- The first theme in the fallback hierarchy is the theme that you assign through the admin panel (in `System > Configuration > Design`). The last theme in hierarchy is the "default" theme in "base" design package.
- If the requested file is not found in your custom sub-theme, Magento will try to find that file in the "default" theme from the package. If the file is not found, Magento will call the file from the "base" package.
- The directory structure inside the sub-theme must replicate the directory structure of the default theme.

6. Theme features and elements

In this chapter you can find description and configuration details of selected elements of this theme, such as home page, built-in static blocks, CSS classes etc.

> **Note:**
>
> The list of all settings available in the theme admin panel in `System > Configuration` can be found in the next chapters.

6.1 Home page

After the theme is installed, you can import sample versions of the home page. For more details about import refer to **7.12 Data Import**.

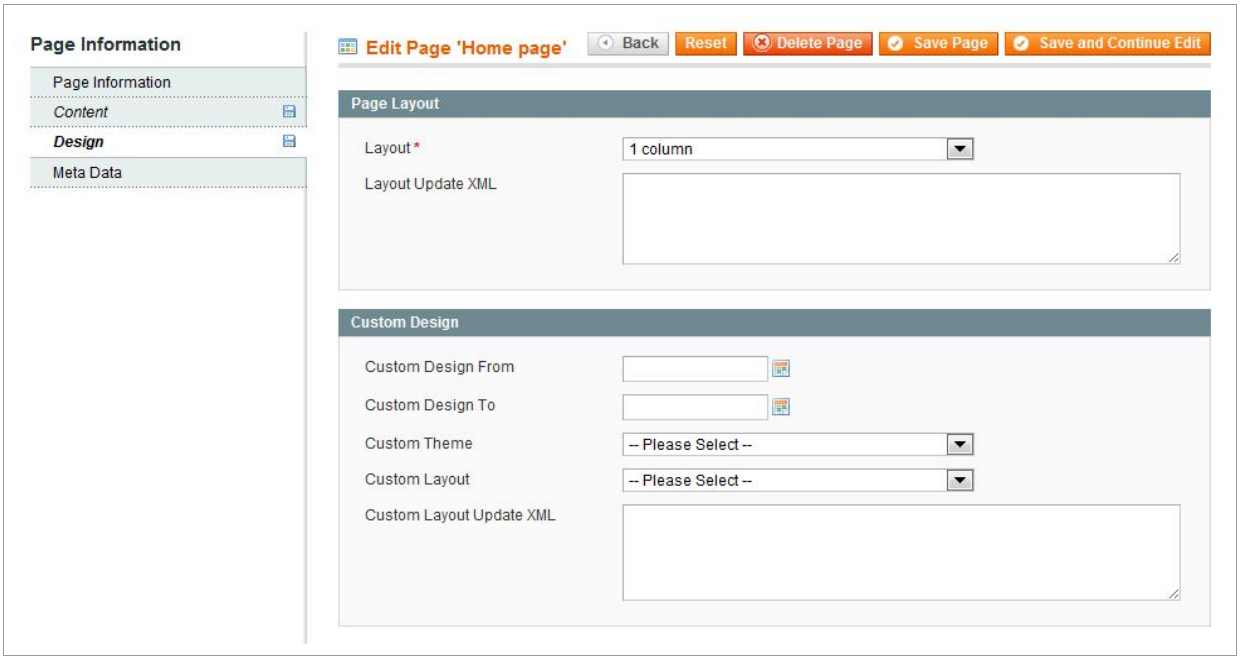
To manage pages in Magento, go to *CMS > Pages* section. By default in Magento a page with **URL Key** `home` is displayed as the store's home page. One of a few sample home pages which you can import, has the following URL key: `ultimo-home-page`. So to display that page as a home page in your store, you need to change its URL key to simple `home`.

> **Note**:
>
> if the page with that URL key already exists (which is true in most of Magento installations) you will need to delete existing page or change its URL key.

Alternatively you can indicate which CMS page you want to use as your home page. To do that, go to *System > Configuration > General > Web > Default Pages* tab and select the desired page in the **CMS Home Page** field.

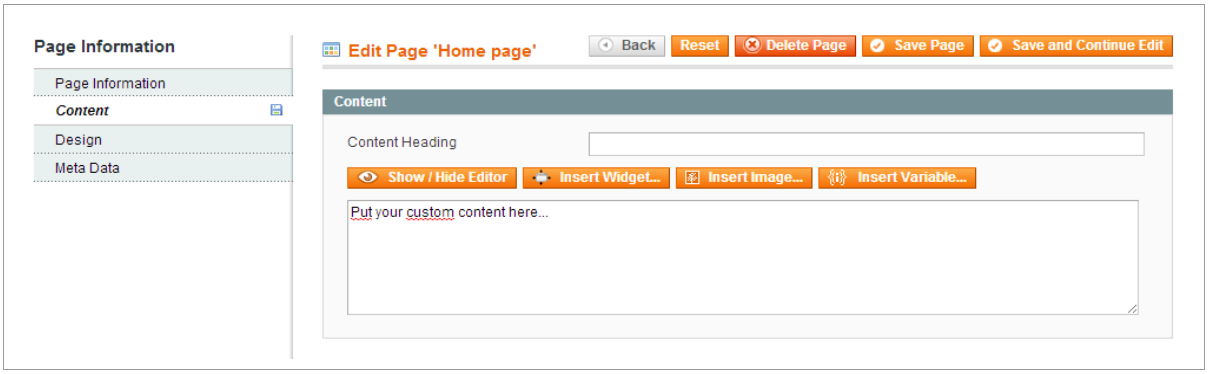
6.1.1 Home page layout

For every CMS page in Magento you can select page layout - the number of columns: one, two, or three columns. To change the layout of the home page, go to *CMS > Pages*, select the page, select one of the options in the **Layout** field and click **Save Page** button:



6.1.2 Home page content

To edit the content of the home page, go to `CMS > Pages` and open the active home page. Content can be found in the main field of the **Content** tab:



Important:

always turn off WYSIWYG editor when you want to edit HTML or Magento's CMS tags in the page/block content. Otherwise WYSIWYG editor will break the content. Refer to **4.8.4 WYSIWYG editor** for more details.

6.1.3 Home page blocks

The following chapter describes blocks which can be displayed inside the content area of the home page. You can enable each block by inserting the block code (CMS markup tag) into the content field.

> **Important:**
>
> always turn off WYSIWYG editor when you want to edit HTML or Magento's CMS tags in the page/block content. Otherwise WYSIWYG editor can break the content. Refer to **4.8.4 WYSIWYG editor** for more details.

There are also special static blocks which can be displayed inside the sidebars of the home page. For more details refer to **6.2.5 Home page**.

6.1.3.A Image slideshow

You can display the slideshow at the top of the home page. Refer to **11. Slideshow** for more details.

6.1.3.B Banners

Here you can see some examples of banners. Banners can be placed on any CMS page, also on the home page. Each banner is a simple image inside a link.

By default images are uploaded to `media/wysiwyg/infortis/ultimo/custom/banners/` directory but you can put them in any other place inside `media` directory. You can upload images via FTP or by using Magento's WYSIWYG editor. WYSIWYG editor will create a HTML tag with correct path to the image, for example:

```
<img src="{{media url="wysiwyg/infortis/ultimo/custom/payment.gif"}}" alt="" />
```

Banners are placed inside `<div>` tags with grid classes. Here you can read more about how to use grid classes on your custom pages and blocks: **6.4 Grid system**.

Example of 3 equal banners:

```
<div class="nested-container">
    <div class="page-banners grid-container-spaced">
        <div class="grid12-4 banner">
            <a href="{{store direct_url="about-magento-demo-store"}}" title="Click me">
                <img src="{{media url="wysiwyg/infortis/ultimo/custom/banners/01.png"}}"
alt="Sample banner" />
            </a>
        </div>
        <div class="grid12-4 banner">
            <a href="{{store direct_url="about-magento-demo-store"}}" title="Click me">
                <img src="{{media url="wysiwyg/infortis/ultimo/custom/banners/02.png"}}"
alt="Sample banner" />
            </a>
        </div>
        <div class="grid12-4 banner">
            <a href="{{store direct_url="about-magento-demo-store"}}" title="Click me">
                <img src="{{media url="wysiwyg/infortis/ultimo/custom/banners/03.png"}}"
alt="Sample banner" />
            </a>
        </div>
    </div>
</div>
```

Example of 3 banners (not equal):

```
<div class="nested-container">
    <div class="page-banners grid-container-spaced">
        <div class="grid12-3 banner">
            <a href="{{store direct_url="about-magento-demo-store"}}" title="Click me">
                <img src="{{media url="wysiwyg/infortis/ultimo/custom/banners/11.png"}}"
alt="Sample banner" />
            </a>
        </div>
        <div class="grid12-3 banner">
            <a href="{{store direct_url="about-magento-demo-store"}}" title="Click me">
                <img src="{{media url="wysiwyg/infortis/ultimo/custom/banners/11.png"}}"
alt="Sample banner" />
            </a>
        </div>
        <div class="grid12-6 banner">
            <a href="{{store direct_url="about-magento-demo-store"}}" title="Click me">
                <img src="{{media url="wysiwyg/infortis/ultimo/custom/banners/03.png"}}"
alt="Sample banner" />
            </a>
        </div>
    </div>
```

```
</div>
```

6.1.3.C Products grid

To display a grid of products similar to the grid which is displayed in category view, add this block code to your home page content:

```
{{block type='catalog/product_list' category_id='4' grid_column_count='4' hide_toolbar='1'
template='catalog/product/list.phtml'}}
```

Value of **grid_column_count** parameter determines the number of products displayed in a single row. This parameter is an equivalent of the option *Number of Columns* which can be found in *System > Configuration > Theme Settings > Category View (Grid Mode)* section (refer to chapter **7.3 Category View (Grid Mode)** for more details).

6.1.4 Featured Products Slider

Featured Products Slider can display products from any category (only single category is allowed). To display the slider, enter the following code in the content field of any page or static block, Magento will automatically replace the code with a slider during page rendering:

```
{{block type="ultimo/product_list_featured"
template="catalog/product/list_featured_slider.phtml" category_id="3" product_count="12"
hide_button="1" block_name="My Products"}}
```

Note:

Make sure that in the **category_id** parameter you specified an identifier of the category which actually exists in your store (and which is active) – otherwise the block won't appear. Other parameters are described below.

Important:

in order to display the slider properly, you may need to reindex Magento (**4.2 Index management**) and flush the cache (**4.1 Caching system**) after making changes to the catalog (adding or editing categories, products, attributes etc.).

Parameters

Global settings related to product sliders can be found in **7.8 Product sliders**. Global settings can be overridden per slider with the following parameters:

Required parameters:

- category_id – identifier of the category which you want to present in the slider. Make sure you enter identifier of the category which actually exists in your store (and which is active) – otherwise the block won't appear.

- **product_count** – the total number of products presented in the slider.
- **block_name** – the name of the block displayed as a heading.

Optional parameters:

- **breakpoints** – with this parameter you can specify the number of products in a row visible with a particular browser viewport width. Use the following format (this is the default value, it will be used if this parameter is not specified in the block code):

```
breakpoints="[0, 1], [320, 2], [480, 3], [768, 4], [960, 5], [1280, 6]"
```

  Value of this parameter consists of pairs of numbers in brackets separated by comma. In each pair of numbers `[A, B]` number `A` indicates the web browser viewport width, number `B` indicates the number of products visible if browser viewport width is greater that the width specified in `A`.
  In other words: display `B` products if viewport width is above `A` pixels. Below you can find some examples:

  - **Example 1:**

```
breakpoints="[0, 2], [480, 3], [960, 5]"
```

    which can be translated as: display 2 products if viewport width is between 0px and 480px, display 3 products if viewport width is above 480px, display 5 products if viewport width is above 960px.

  - **Example 2:**

```
breakpoints="[0, 4]"
```

    which can be translated as: always display 4 products.

  - **Example 3:**

```
breakpoints="[0, 3], [960, 6]"
```

    which can be translated as: display 3 products on narrow screens (if viewport width is below 960px), and display 6 products on wider screens (above 960px).

- **is_responsive** – enter `0` if the slider **should not be responsive**. If this parameter is not specified, it is always assumed that the slider is responsive.

> **Important:**
>
> If the slider **is not** responsive, the number of visible products should be specified with the additional **show_items** parameter (see below).

  - ***show_items*** – the number of products in a row in the non-responsive slider. Use this parameter only if responsive behavior was disabled with parameter **is_responsive** (see above)

- **timeout** – to animate the slider automatically, specify the time (in milliseconds) between transitions. Note that 1000 milliseconds = 1 second.
- **move** – number of products that should move on animation. Allowed values:
  - enter `1` to move one product.
  - enter `0` to move all visible products. This is also the default value which is used if the parameter is not specified.
- **loop** – enter `1` if the slider should loop.
- **pagination** – enter `1` to display pagination.
- **img_width** – enter product image width.
- **img_height** – enter product image height. Do not specify this parameter if you want to keep the aspect ratio of the product images – if height is not specified, it will be calculated automatically based on width.
- **centered** – enter `1` to align elements of the slider to the center.
- **size** – with this parameter you can decrease the size of the elements of the slider, such as product name, button, product labels. Available values:
  - `size-s` – elements a little bit smaller than the standard size
  - `size-xs` – elements much smaller than the standard size
- **is_random** – enter `1` to display random products from the selected category. Otherwise enter `0` (in this case you can also omit this parameter, because `0` is the default value).

- **hide_button** – enter `1` to hide “Add to cart” button (by default displayed below each product).
- **sort_by** – with this parameter you can change the order of products. Available values:
  - `position` – default order of products in category
  - `name` – sort by name
  - `price` – sort by price
- **sort_direction** – with this parameter you can change the direction of sorting. Available values:
  - `ASC` – ascending
  - `DESC` – descending

Display any set of products using the slider

You can use Featured Products Slider to display any set of products in your store. Slider can display products **from any category** so you can simply create categories which will group any type of products you need, for example: bestsellers, products from specific brand, new products, etc.

You can, for example, display “on sale” products on the home page:

1. Create a new category. Name it "Special Offer", for example. You can also select **No** in **Include in Navigation Menu** field – thanks to that category will not be displayed in the main menu.
2. Add all products with special price (or any other products) to "Special Offer" category.
3. You might need to **reindex the data** in *System > Index Management* after adding products to the new category.
4. Display "Special Offer" category using Featured Product Slider:
   a) Specify the ID of the "Special Offer" category in the block code. Make sure you use category ID which really exists in your store – otherwise the block won't appear.

b) Specify the number of products in the corresponding parameter.

c) Add the block code to the home page content.

5. Flush Magento cache.

Examples

Here you can see some examples of the slider block code:

```
{{block type="ultimo/product_list_featured"
template="catalog/product/list_featured_slider.phtml" category_id="3" product_count="12"
breakpoints="[0, 1], [320, 2], [480, 3], [768, 4], [960, 5], [1280, 6]" pagination="1"
centered="1" hide_button="1" block_name="Our Featured Products"}}
```

```
{{block type="ultimo/product_list_featured"
template="catalog/product/list_featured_slider.phtml" category_id="4" product_count="8"
size="size-xs" is_random="1" block_name="Random Products"}}
```

```
{{block type="ultimo/product_list_featured"
template="catalog/product/list_featured_slider.phtml" category_id="3" product_count="16"
is_responsive="0" show_items="5" hide_button="1" block_name="Non-responsive Slider"}}
```

6.1.5 New Products Slider

New Products Slider can display products which are marked as new in your store. To mark the product as "new", go to `Catalog > Manage Products`, select the product and in **Set Product as New from Date** and **Set Product as New to Date** fields, enter the date range in which the product will be promoted as a new product.

To display the slider, enter the following code in the content field of any page or static block, Magento will automatically replace the code with a slider during page rendering:

```
{{block type="catalog/product_new" template="catalog/product/new.phtml" products_count="10"
hide_button="1" block_name="My New Products"}}
```

Important:

In order to display this block properly, you may need to reindex Magento (**4.2 Index management**) and flush the cache (**4.1 Caching system**) after marking products as new and after making changes to the catalog (adding or editing categories, products, attributes etc.).

Parameters

Global settings related to product sliders can be found in **7.8 Product sliders**. Global settings can be overridden per slider with the following parameters:

Required parameters:

- products_count – the total number of products presented in the slider. Note the letter *"s"* at the end of *"products"*.
- block_name – the name of the block displayed as a heading.

Optional parameters:

- breakpoints – with this parameter you can specify the number of products in a row

visible with a particular browser viewport width. Use the following format (this is the default value, it will be used if this parameter is not specified in the block code):

```
breakpoints="[0, 1], [320, 2], [480, 3], [768, 4], [960, 5], [1280, 6]"
```

Value of this parameter consists of pairs of numbers in brackets separated by comma. In each pair of numbers `[A, B]` number `A` indicates the web browser viewport width, number `B` indicates the number of products visible if browser viewport width is greater that the width specified in `A`.
In other words: display `B` products if viewport width is above `A` pixels. Below you can find some examples:

- **Example 1:**

```
breakpoints="[0, 2], [480, 3], [960, 5]"
```

which can be translated as: display 2 products if viewport width is between 0px and 480px, display 3 products if viewport width is above 480px, display 5 products if viewport width is above 960px.

- **Example 2:**

```
breakpoints="[0, 4]"
```

which can be translated as: always display 4 products.

- **Example 3:**

```
breakpoints="[0, 3], [960, 6]"
```

which can be translated as: display 3 products on narrow screens (if viewport width is below 960px), and display 6 products on wider screens (above 960px).

- `is_responsive` – enter `0` if the slider **should not be responsive**. If this parameter is not specified, it is always assumed that the slider is responsive.

> **Important:**
>
> If the slider **is not** responsive, the number of visible products should be specified with the additional **show_items** parameter (see below).

  - ***show_items*** – the number of products in a row in the non-responsive slider. Use this parameter only if responsive behavior was disabled with parameter **is_responsive** (see above)

- **timeout** – to animate the slider automatically, specify the time (in milliseconds) between transitions. Note that 1000 milliseconds = 1 second.
- **move** – number of products that should move on animation. Allowed values:
  - enter `1` to move one product.
  - enter `0` to move all visible products. This is also the default value which is used if the parameter is not specified.
- **loop** – enter `1` if the slider should loop.
- **pagination** – enter `1` to display pagination.
- **img_width** – enter product image width.
- **img_height** – enter product image height. Do not specify this parameter if you want to keep the aspect ratio of the product images – if height is not specified, it will be calculated automatically based on width.
- **centered** – enter `1` to align elements of the slider to the center.
- **size** – with this parameter you can decrease the size of the elements of the slider, such as product name, button, product labels. Available values:
  - `size-s` – elements a little bit smaller than the standard size
  - `size-xs` – elements much smaller than the standard size
- **hide_button** – enter `1` to hide “Add to cart” button (by default displayed below each product).
- *init_delay* – this parameter is no longer supported.

Examples

Here you can see some examples of the block code:

```
{{block type="catalog/product_new" template="catalog/product/new.phtml" products_count="10"
centered="1" size="size-s" hide_button="1" block_name="New Products"}}
```

```
{{block type="catalog/product_new" template="catalog/product/new.phtml" products_count="20"
breakpoints="[0, 1], [320, 2], [480, 3], [768, 4], [960, 5], [1280, 6]" hide_button="1"
block_name="New products in our store"}}
```

6.2 Built-in CMS blocks

There are more than 40 predefined CMS blocks (static blocks) – content placeholders displayed in almost every part of the store, ready to show any content you need: text or HTML. You can import all of these blocks with one click – refer to **7.12 Data Import** for more details about the import.

> **Important:**
>
> content of all static blocks is also available in XML format in case you need to restore the original content of selected blocks. XML can be found in the following directory:
>
> `app/code/local/Infortis/Ultimo/etc/import/`
>
> Refer to **7.12 Data Import** for more details.

> **Important:**
>
> always turn off WYSIWYG editor when you want to edit HTML or Magento's CMS tags in the page/block content. Otherwise WYSIWYG editor will break the content. Refer to **4.8.4 WYSIWYG editor** for more details.

After importing, some of the blocks are disabled. You will need to enable them in order to show their content in your store. You can replace the default content of each block with your own content.

> **Note:**
>
> in order to see your changes, you need to refresh Magento cache after making any changes in your static blocks.

For more information about static block management in Magento refer to chapter **4.8.2 Static blocks**.

Identifiers of all built-in blocks are described later in this chapter. For example, `block_footer_payment` is an identifier of the block which displays payment information in the footer section. So if you enable that static block, the content of that block will be automatically displayed in the footer of your store.

Sample content

Most of the blocks already contain sample content. You can replace it with any custom content, but using provided samples will help you to keep the design consistent. For example, in all blocks containing links you can use the following HTML template:

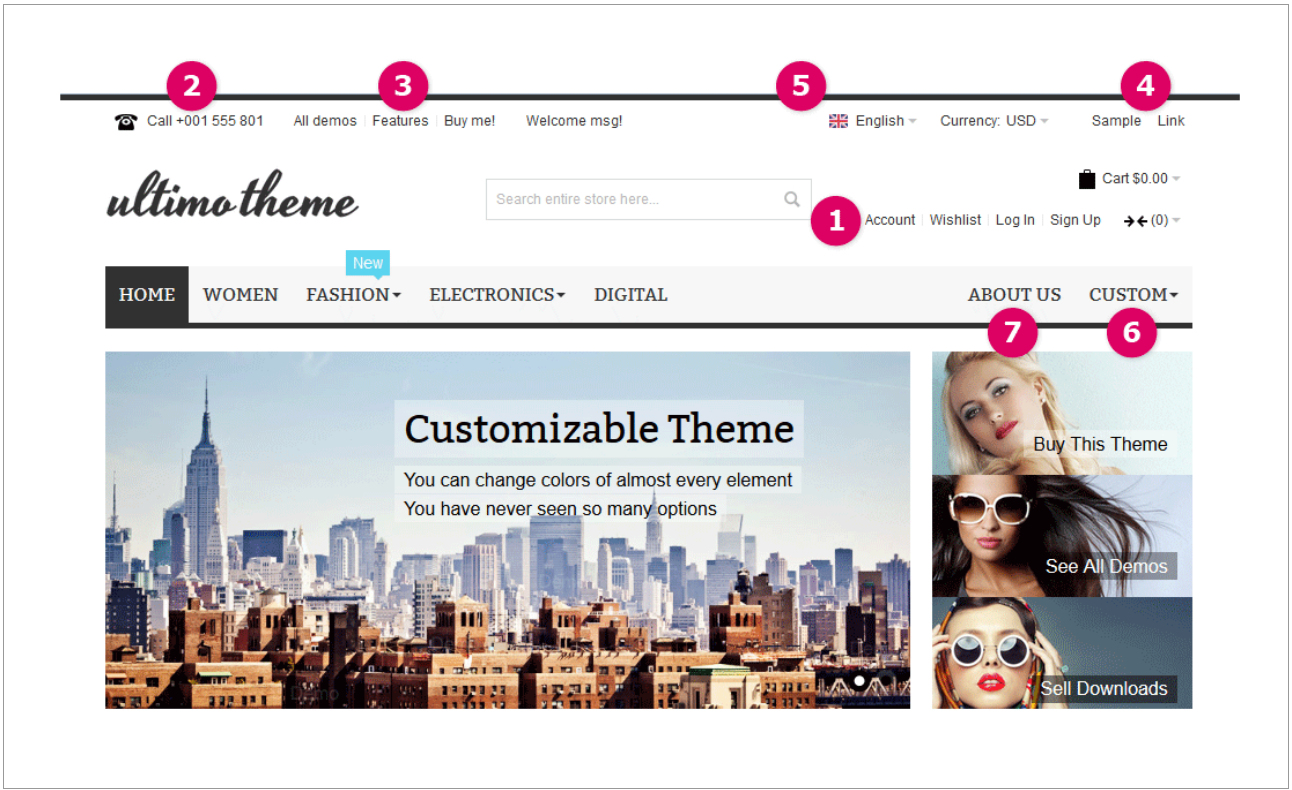
```
<ul class="links">
     <li class="first">
             <a href="{{store url="path/to/page"}}">First Sample Link</a>
     </li>
     <li>
             <a href="{{store url="path/to/page"}}">Second Sample Link</a>
     </li>
     <li class="last">
             <a href="{{store url="path/to/page"}}">Last Sample Link</a>
     </li>
</ul>
```

Replace `path/to/page` with your own value. It will be appended to the store's base URL with additional slash (`/`) at the end of the resulting URL.

Below you can find a list of identifiers of all built-in static blocks available in this theme.

6.2.1 Header

The following block identifiers can be used to display blocks in the header section of the store:



1. block_header_top_links – this block can substitute Magento's default Top Links (set of links: *Account*, *Wishlist*, *Log In*). Default Top Links can be disabled in the admin panel, refer to **7.9 Default Magento blocks** for more details. There's also another static block block_header_top_links2 in this area but it will be removed in the future updates as it duplicates block_header_top_links2.

2. block_header_top_left – block at the top of the header, floating left

3. block_header_top_left2 – block at the top of the header, floating left

4. block_header_top_right – block at the top of the header, floating right

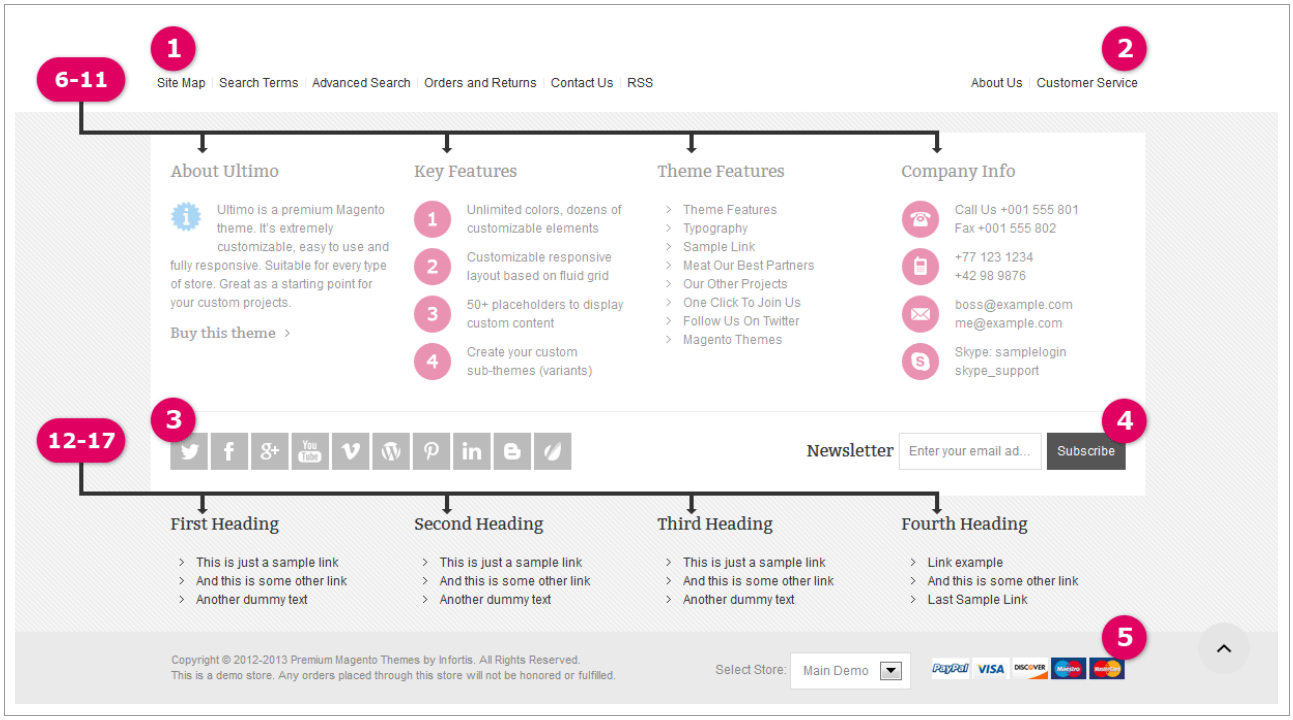
5. block_header_top_right2 – block at the top of the header, floating right (at the left side of language and currency switchers). In version 1.5.0 it replaced block_header_top_right

6. block_header_nav_dropdown – drop-down box in the main menu. This block is deprecated and will be removed in future versions. There is a better method to add custom drop-down boxes to the menu, for more details refer to **13.4 Custom drop-down boxes in the menu**.

7. block_nav_links – custom links in the main menu. This block is deprecated and will be removed in future versions. There is a better method to add custom links to the menu, for more details refer to **13.3 Custom links in the menu**.

6.2.2 Footer

The following block identifiers can be used to display blocks in the footer section of the store:


6-11
1
Site Map | Search Terms | Advanced Search | Orders and Returns | Contact Us | RSS
2
About Us | Customer Service
About Ultimo
Ultimo is a premium Magento theme. It's extremely customizable, easy to use and fully responsive. Suitable for every type of store. Great as a starting point for your custom projects.
Buy this theme
Key Features
1 Unlimited colors, dozens of customizable elements
2 Customizable responsive layout based on fluid grid
3 50+ placeholders to display custom content
4 Create your custom sub-themes (variants)
Theme Features
Theme Features
Typography
Sample Link
Meat Our Best Partners
Our Other Projects
One Click To Join Us
Follow Us On Twitter
Magento Themes
Company Info
Call Us +001 555 801
Fax +001 555 802
+77 123 1234
+42 98 9876
boss@example.com
me@example.com
Skype: samplelogin
skype_support
12-17
3
4
Newsletter
Enter your email ad...
Subscribe
First Heading
This is just a sample link
And this is some other link
Another dummy text
Second Heading
This is just a sample link
And this is some other link
Another dummy text
Third Heading
This is just a sample link
And this is some other link
Another dummy text
Fourth Heading
Link example
And this is some other link
Last Sample Link
5
Copyright © 2012-2013 Premium Magento Themes by Infortis. All Rights Reserved.
This is a demo store. Any orders placed through this store will not be honored or fulfilled.
Select Store:
Main Demo

1. block_footer_links – this block can substitute Magento's default Footer Links (set of links: *Site Map*, *Search Terms*, *Contact Us* etc.). Default Footer Links can be disabled in the admin panel, refer to **7.9 Default Magento blocks** for more details.

2. block_footer_links2 – additional block for footer links, floating right.

3. block_footer_primary_bottom_left – icons with links to social services.

4. block_footer_primary_bottom_right – this block can substitute Magento's default newsletter subscription form. Newsletter form can be disabled in the admin panel, refer to **7.9 Default Magento blocks** for more details.

5. block_footer_payment – information about available payment methods, credit cards, SSL certificates etc.

6. block_footer_column1 – a column of links in the main area of the footer section, suitable for any custom content: social services links, shipping information,

promotions, company address etc. You can use up to six similar static blocks.

7. **block_footer_column2** – see above
8. **block_footer_column3** – see above
9. **block_footer_column4** – see above
10. **block_footer_column5** – see above
11. **block_footer_column6** – see above
12. **block_footer_row2_column1** – additional column of links in the secondary area of the footer section, suitable for any custom content: social services links, shipping information, promotions, company address etc. You can use up to six similar static blocks.
13. **block_footer_row2_column2** – see above
14. **block_footer_row2_column3** – see above
15. **block_footer_row2_column4** – see above
16. **block_footer_row2_column5** – see above
17. **block_footer_row2_column6** – see above

6.2.3 Category view

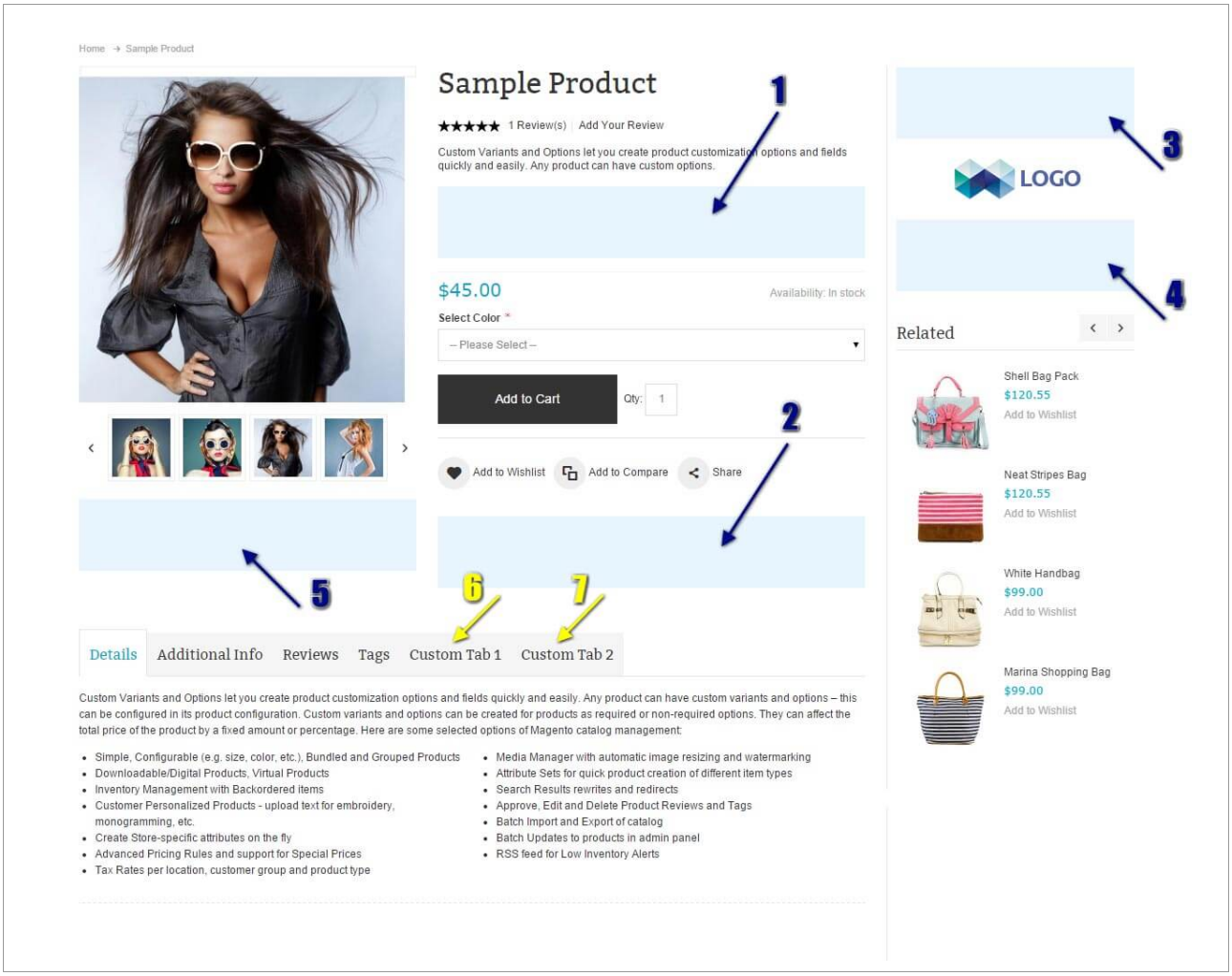
The following block identifiers can be used to display blocks in the category view:

1. **block_category_above_collection** – block above the products listing in the category view.
2. **block_category_above_empty_collection** – block above the products listing, displayed if there are no products in the category.
3. **block_category_below_collection** – block below the products listing in the category view.

> **Note:** in the category view you can also display a block with a list of all categories. For more info about that block refer to **13.1.4 Sidebar Menu**.

6.2.4 Product page

The screenshot below presents positions of block containers which are available on product page.


Home → Sample Product
Sample Product
1 Review(s) Add Your Review
Custom Variants and Options let you create product customization options and fields quickly and easily. Any product can have custom options.
$45.00
Availability: In stock
Select Color *
– Please Select –
Add to Cart
Qty: 1
Add to Wishlist
Add to Compare
Share
LOGO
Related
Shell Bag Pack
$120.55
Add to Wishlist
Neat Stripes Bag
$120.55
Add to Wishlist
White Handbag
$99.00
Add to Wishlist
Marina Shopping Bag
$99.00
Add to Wishlist
1
2
3
4
5
6
7
Details
Additional Info
Reviews
Tags
Custom Tab 1
Custom Tab 2
Custom Variants and Options let you create product customization options and fields quickly and easily. Any product can have custom variants and options – this can be configured in its product configuration. Custom variants and options can be created for products as required or non-required options. They can affect the total price of the product by a fixed amount or percentage. Here are some selected options of Magento catalog management:
Simple, Configurable (e.g. size, color, etc.), Bundled and Grouped Products
Downloadable/Digital Products, Virtual Products
Inventory Management with Backordered items
Customer Personalized Products - upload text for embroidery, monogramming, etc.
Create Store-specific attributes on the fly
Advanced Pricing Rules and support for Special Prices
Tax Rates per location, customer group and product type
Media Manager with automatic image resizing and watermarking
Attribute Sets for quick product creation of different item types
Search Results rewrites and redirects
Approve, Edit and Delete Product Reviews and Tags
Batch Import and Export of catalog
Batch Updates to products in admin panel
RSS feed for Low Inventory Alerts

You can assign any number of **custom static blocks** to each container. For details about assigning static blocks to containers refer to chapter **4.8.1 Add blocks to selected pages**.

Containers are suitable for any kind of additional information:
discounts and promotions, available shipping methods, available payment methods, promotional image banners, social services bookmarks (such as AddThis bookmarks) etc.

Here is a list of containers which are available on product page:

1. **Product View, Primary Column, Container 1** – container displayed below product's short description inside the primary column.

2. **Product View, Primary Column, Container 2** – container at the bottom of the primary column.

> **Note:**
> for backward compatibility, the static block with ID ***block_product_primary_bottom*** (if it exists in your Magento) is displayed automatically inside this container.

3. **Product View, Secondary Column, Container 1** – container at the top of the secondary column.

4. **Product View, Secondary Column, Container 2** – container at the bottom of the secondary column.

> **Note:**
> for backward compatibility, the static block with ID ***block_product_secondary_bottom*** (if it exists in your Magento) is displayed automatically inside this container.

5. **Product View, Image Column, Below Images** – container displayed just below the image gallery.

> **Note:**
> for backward compatibility, the static block with ID ***block_product_img_bottom*** (if it exists in your Magento) is displayed automatically inside this container.

Additionally you can also display two static blocks as tabs on product page.

6. **block_product_tab1** – to display custom tab on product page, create a static block with identifier `block_product_tab1`. The content of that block will be displayed inside the panel of the tab and the title of that block will be displayed as the name of the tab.

7. block_product_tab2 – to display second custom tab on product page, create a static block with identifier `block_product_tab2`. The content of that block will be displayed inside the panel of the tab and the title of that block will be displayed as the name of the tab.

> **Note:**
>
> content of these tabs will be displayed on the product page of every product in the store.

There are two other tabs on product page for **product-specific** info:

- **Details** – tab which displays product description. You can add there any custom content, it can be a simple text as well as a complex HTML with images, lists etc. You can also embed videos from services such as YouTube, Vimeo etc.

- **Additional Info** – tab which displays product-specific attributes. You can enable/disable each attribute's visibility on product page with **Visible on Product View Page on Front-end** field in *Catalog > Attributes > Manage Attributes > [select attribute]*.

6.2.5 Home page

The following block identifiers can be used to display blocks on the home page:

1. block_home_preface – block at the top of the home page, above sidebars and main content area.
2. block_home_leftl – block at the top of the left sidebar on the home page. Blocks in the left sidebar of the home page are **always displayed above** any other blocks which are currently displayed in that sidebar.
3. block_home_left2 – see above
4. block_home_left3 – see above
5. block_home_rightl – block at the top of the right sidebar on the home page. Blocks in the right sidebar of the home page are **always displayed above** any other blocks which are currently displayed in that sidebar.
6. block_home_right2 – see above
7. block_home_right3 – see above
8. block_home_postscript – block at the bottom of the home page, below sidebars and main content area.

6.2.6 Shopping cart

The following block identifiers can be used to display blocks in the shopping cart:

1. block_cart_below_table – block below the main table in the shopping cart.
2. block_cart_below_totals – block below the shopping cart totals.

6.2.7 Cart drop-down

The following block identifiers can be used to display blocks in the cart drop-down box:

1. block_mini_cart_above_products – block inside cart drop-down, suitable for a short info about promotions, sales etc.

6.2.8 One page checkout

The following block identifiers can be used to display blocks in the one page checkout:

1. block_opc_below_progress – block below the checkout progress block.

6.2.9 Left column

The following block identifiers can be used to display blocks in the left sidebar. Blocks will be visible on all pages which contain the left sidebar, e.g. in category view.

1. block_left_top – block at the top of the left sidebar.
2. block_left_bottom – block at the bottom of the left sidebar.

6.2.10 Right column

The following block identifiers can be used to display blocks in the right sidebar. Blocks will be visible on all pages which contain the right sidebar, e.g. on Advanced Search page.

1. block_right_top – block at the top of the right sidebar.

2. block_right_bottom – block at the bottom of the right sidebar.

6.2.11 CMS pages

The following block identifiers can be used to display blocks on the CMS pages (excluding home page):

1. block_cms_page_top – block at the top of any CMS page (excluding home page).

6.3 Pages and elements

6.3.1 Product page

6.3.1.A Social bookmarks (AddThis)

To display social bookmarks this theme uses the AddThis widget. The code of the widget can be found on this site: *http://www.addthis.com/get*

You can generate many different versions of the widget code to create customized bookmarks. Then, add the code to your static block or page and **flush Magento cache**.

AddThis bookmarks can be, for example, placed in the static block with ID `block_product_secondary_bottom`. This block is positioned at the bottom of the secondary column on product page.

> **Important:**
>
> always turn off WYSIWYG editor when you want to edit HTML or Magento's CMS tags in the page/block content. Otherwise WYSIWYG editor will break the content. Refer to **4.8.4 WYSIWYG editor** for more details.

6.3.2 Drop-down cart

This theme is using Magento's default sidebar cart to display drop-down cart at the top of the page. In order to show drop-down cart, option **Display Shopping Cart Sidebar** in *System > Configuration > Checkout > Shopping Cart Sidebar* has to be enabled.

6.4 Grid system

Theme is based on 12-column grid system. This means that for every element you can specify width in grid units (from 1 unit to 12 units) using grid classes. These are available classes:

```
grid12-1
grid12-2
grid12-3
grid12-4
grid12-5
grid12-6
grid12-7
grid12-8
grid12-9
grid12-10
grid12-11
grid12-12
```

The number at the end of the class name indicates number of grid units. For example, class `grid12-3` will create a block which is 3-units wide.

As this is a 12-column grid, each grid class spans a number of those 12 columns, and should always add up to 12 for each row.

6.4.1 Simple grid examples

2 columns

If you want to display content on your custom page in two equal columns, create two blocks (`div`) and add appropriate class with number of grid units. In this case use `grid12-6` class to create two 6-units wide blocks. And 6 + 6 equals 12, so you will create a row of two equal columns.

Then wrap the blocks inside another block with class `grid-container` – this class is important to clear floats of the columns (all columns float to the left side) so that the content of the grid doesn't overlap with other content below and above the grid. Here's an example:

```
<div class="grid-container show-grid">
     <div class="grid12-6">Sample text...</div>
     <div class="grid12-6">Sample text...</div>
</div>
```

When developing the grid, you can also temporarily add class `show-grid` to visualize the grid (like in the above example).

3 columns

```
<div class="grid-container">
     <div class="grid12-4">Sample text...</div>
     <div class="grid12-4">Sample text...</div>
     <div class="grid12-4">Sample text...</div>
</div>
```

4 columns

```
<div class="grid-container">
     <div class="grid12-3">Sample text...</div>
     <div class="grid12-3">Sample text...</div>
     <div class="grid12-3">Sample text...</div>
     <div class="grid12-3">Sample text...</div>
</div>
```

3 columns (not equal)

Of course columns doesn't have to be equal, you can use grid units in different proportions. But the sum of all units in one row should be equal to 12, otherwise some columns may drop to another row. Here is an example of three columns which are not equal:

```
<div class="grid-container">
     <div class="grid12-7">Sample text...</div>
     <div class="grid12-3">Sample text...</div>
     <div class="grid12-2">Sample text...</div>
</div>
```

6.4.2 Multiple rows

To create multiple rows of columns simply add more units to the container:

```
<div class="grid-container">
     <div class="grid12-6">Sample text...</div>
     <div class="grid12-6">Sample text...</div>
     <div class="grid12-4">Sample text...</div>
     <div class="grid12-4">Sample text...</div>
     <div class="grid12-4">Sample text...</div>
</div>
```

Space between rows

To add even vertical spaces between rows, replace `classgrid-container` with class `grid-container-spaced`:

```
<div class="grid-container-spaced">
     <div class="grid12-6">Sample text...</div>
     <div class="grid12-6">Sample text...</div>
     <div class="grid12-4">Sample text...</div>
     <div class="grid12-4">Sample text...</div>
     <div class="grid12-4">Sample text...</div>
</div>
```

6.4.3 Grid on mobile devices

If the web browser's viewport width is **below 768 px**, all grid columns are stretched (they are as wide as the container) and stack vertically. If you want to avoid this and want your columns to stay narrow on narrow screens, you can add special class `mobile-grid` to grid classes. It can be added to the following grid classes: `grid12-1`, `grid12-2`, `grid12-3`.

On narrow screens columns could be too narrow to display content properly, so thanks to the `mobile-grid` class, each column will be 2 times wider than the actual number of units in the class name, i.e. `grid12-3` will behave like `grid12-6`.

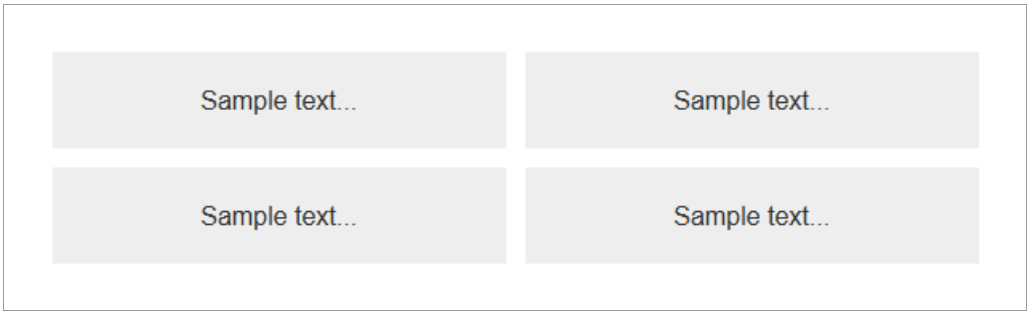
When the viewport width goes **below 480 px**, all grid columns with the `mobile-grid` class will also be stretched and stacked vertically, just like all the other columns.

The following example will create 4 equal columns:

```
<div class="grid-container-spaced">
     <div class="grid12-3 mobile-grid">Sample text...</div>
     <div class="grid12-3 mobile-grid">Sample text...</div>
     <div class="grid12-3 mobile-grid">Sample text...</div>
     <div class="grid12-3 mobile-grid">Sample text...</div>
</div>
```

which will look like this:

When the viewport is below 768 px, the 4 columns will be automatically transformed to 2 rows of 2 columns:



6.5 CMS components

In the following chapter you can find description of selected HTML components which can be used in CMS pages and static blocks.

6.5.1 CSS utility classes

Here you can find description of some useful CSS classes which can be used in CMS pages and static blocks.

6.5.1.A Hide elements for specific screen resolution

Use these classes to hide elements below specific screen resolution. Number at the end of the class name indicates the resolution:

```
hide-below-1680
hide-below-1440
hide-below-1360
hide-below-1280
hide-below-960
hide-below-768
hide-below-480
hide-below-320
```

For example, add class `hide-below-1280` to the HTML element which needs to be hidden if the screen is less than 1280 pixels wide.

```
<div class="hide-below-1280">
     This is just a sample text
</div>
```

6.5.2 Collapsible blocks

With collapsible block, your content can be shown and hidden on button click, you can build components similar to accordions. Blocks like this are used in the footer static blocks in the demo:

```
<div class="collapsible">
     <h6 class="block-title active">My block title</h6>
     <div class="block-content">Sample content</div>
</div>
```

If you remove `active` class, the block will be collapsed by default.

If you want the block to be collapsed **only** on mobile devices (with viewport width below 768 pixels), use `mobile-collapsible` class instead of `collapsible` class in the main wrapper:

```
<div class="mobile-collapsible">
     <h6 class="block-title">My block title</h6>
     <div class="block-content">Sample content</div>
</div>
```

6.5.3 Icons

6.5.3.A Font icons

You can place icons just about anywhere using simple markup. We are going to use an inline HTML element such as `<span>` and add appropriate classes to it. These are required classes: `ic` and the icon's name prefixed with `ic-`, for example `ic-star`. Here's the full example of the code which will add star icon:

```
<span class="ic ic-star"></span>
```

> **Note:**
>
> list of names of all available icons can be found in `Resources/Icons_Demo/demo.html` inside the theme package.

Add CSS styles to icons

Now you can start having more fun with icons. For example, if you want to change color of the icon, you could just add **inline styles**: add `style` attribute to the icon element and specify a color. For example:

```
<span class="ic ic-star" style="color:red;"></span>
```

You can add inline styles to icons the same way as to any other HTML elements in a HTML document. The `style` attribute can contain any CSS property, such as font-size, text-shadow etc. List of available CSS properties can be found on *https://developer.mozilla.org/en-US/docs/Web/CSS/Reference*

Size

If you change the font-size of the icon's container, the icon gets bigger. To increase icon size relative to the font-size of the icon container, use the following classes: `ic-lg` (increases the size of the icon by 33%), `ic-2x`, `ic-3x`, `ic-4x`, `ic-5x`, `ic-6x`, `ic-7x`, `ic-8x`.

```
<span class="ic ic-star ic-lg"></span>
<span class="ic ic-gift ic-2x"></span>
<span class="ic ic-twitter ic-3x"></span>
```

To have more control over the size of the icon, you can also change the font size of the icon container by adding an inline style:

```
<span class="ic ic-star" style="font-size:32px;"></span>
```

List icons

Use `ic-ul` and `ic-li` to build unordered lists with icons displayed as bullets.

```
<ul class="ic-ul">
     <li><span class="ic ic-star ic-li"></span>Some text example</li>
     <li><span class="ic ic-tag ic-li"></span>Some text example</li>
     <li><span class="ic ic-arrow-right ic-li"></span>Some text example</li>
     <li><span class="ic ic-letter ic-li"></span>Some text example</li>
     <li><span class="ic ic-bell ic-li"></span>Some text example</li>
</ul>
```

Pulled icons and borders

Use `ic-border` and `pull-left` or `pull-right` for easy pull article icons.

```
<span class="ic ic-alarm ic-3x ic-border pull-left "></span> Some sample text. Lid est laborum
et dolorum fuga. Et harum quidem rerum facilis est et expeditasi distinctio.
```

Spinning icons

Use the `ic-spin` class to get any icon to rotate.

```
<span class="ic ic-reload ic-spin"></span>
```

Rotated and flipped

To rotate and flip icons, use the `ic-rotate-*` and `ic-flip-*` classes.

```
<span class="ic ic-twitter"></span> normal<br>
<span class="ic ic-twitter ic-rotate-90"></span> ic-rotate-90<br>
<span class="ic ic-twitter ic-rotate-180"></span> ic-rotate-180<br>
<span class="ic ic-twitter ic-rotate-270"></span> ic-rotate-270<br>
<span class="ic ic-twitter ic-flip-horizontal"></span> ic-flip-horizontal<br>
<span class="ic ic-twitter ic-flip-vertical"></span> ic-flip-vertical
```

Icons inside buttons

Font icons work great inside buttons and other elements.

```
<a class="button" href="#"><span class="ic ic-user"></span>  My Account</a>
```

6.5.3.B Icons with background color (iconbox)

To display an icon inside a box with background color, add `ib` class to the icon container.

For example:

```
<span class="ic ic-star ib"></span>
```

Background color will be automatically added to the icon container. Default background color and color of the icon can be configured in the admin panel:
*System > Configuration > Theme Design > Colors > **Custom Icons With Background (Iconbox)***

Make sure to leave the `<span>` tag empty. Otherwise the contents of the tag will be displayed together with icon and any additional space can dislocate the icon.

Change color on mouse hover

To change the color on mouse hover over the iconbox, add the `ib-hover` class to the iconbox.

```
<span class="ic ic-star ib ib-hover"></span>
```

Size

To increase iconbox size, use the following classes:
`ib-size-l`, `ib-size-xl`, `ib-size-xxl`, `ib-size-xxxl`. For example:

```
<span class="ic ic-star ib ib-size-xxl"></span>
```

The size of the icon is independent of the iconbox and can be increased with classes which were described earlier in this chapter. For example, add class `ic-2x` to make the icon a little bit bigger:

```
<span class="ic ic-star ic-2x ib ib-size-xxl"></span>
```

Shape

To change the shape of the iconbox, use one of the following classes: `ib-circle`, `ib-rounded`, `ib-square`. For example:

```
<span class="ic ic-star ib ib-square"></span>
```

By default the iconbox is circular. If you prefer square icons, you can wrap multiple icons with an element (it can be `<div>`) with class `ib-wrapper--square`:

```
<div class="ib-wrapper—square">
     <span class="ic ic-star ib"></span>
     <span class="ic ic-heart ib"></span>
     <span class="ic ic-letter ib"></span>
</div>
```

6.5.3.C Social icons

To add icons of social networking services you need to use exactly the same markup as for other icons described earlier in this chapter – name of the service needs to be prefixed with `ic-`. For example:

```
<span class="ic ic-twitter"></span>
```

Note:

> list of names of all available icons can be found in `Resources/Icons_Demo/demo.html` inside the theme package.

Icons can be used in many different places. They can also be wrapped inside links. This is very useful when you want to create links to your profiles on social networking services. Here's an example:

```
<div class="social-links">
     <a href="#">
             <span class="ib ib-hover ic ic-lg ic-twitter"></span>
     </a>
     <a href="#">
             <span class="ib ib-hover ic ic-lg ic-facebook"></span>
     </a>
     <a href="#">
             <span class="ib ib-hover ic ic-lg ic-googleplus"></span>
     </a>
     <a href="#">
             <span class="ib ib-hover ic ic-lg ic-youtube"></span>
     </a>
     <a href="#">
             <span class="ib ib-hover ic ic-lg ic-vk"></span>
     </a>
     <a href="#">
             <span class="ib ib-hover ic ic-lg ic-renren"></span>
     </a>
     <a href="#">
             <span class="ib ib-hover ic ic-lg ic-instagram"></span>
     </a>
</div>
```

Replace `#` with URLs of your profiles.

6.5.3.D Custom image icons

You can also use custom image files to create simple icons. With the following markup: `<span>` tag with classes `ib ic ic-img` and `<img>` tag inside the `<span>` tag, you can quickly create custom icon:

```
<span class="ib ic ic-img">
     <img src="{{media url='wysiwyg/infortis/ultimo/icons/star.png'}}" alt="My icon" />
</span>
```

Image file for the icon can be indicated with Magento's CMS tag which will be replaced with the URL of the image, for example: `{{media url='path/to/image.png'}}`. The path is relative to Magento's `media` directory, so you need o store your images inside that directory.

Recommended size of an image: 30 x 30 pixels. An image can be larger but it will be scaled down to fit the size of the iconbox. So it is better to use images of correct size to keep good quality.

6.5.3.E Letters, numbers and other characters as icons

By adding class `ic-char` to the icon, you can display any characters (such as letters, numbers and other special characters) as icons. The code is almost the same as for standard font icons, but inside the `<span>` tag you need to add a single character. For example:

```
<span class="ic ic-char">1</span>
<span class="ic ic-char">2</span>
<span class="ic ic-char">a</span>
<span class="ic ic-char">B</span>
<span class="ic ic-char">$</span>
<span class="ic ic-char">%</span>
```

6.5.4 Text with icon

With the following code using the `feature` class you can display blocks of text (text paragraphs, links, headings etc.) with an additional icon:

```
<div class="feature">
     <span class="left ic ic-star ib"></span>
     <h6 class="above-heading">Secondary Heading</h6>
     <h3>Block Heading Example</h3>
     <p>This is a paragraph of sample text. Lorem ipsum dolor sit, consectetur adipiscing elit.
Etiam neque velit, blandit sed scelerisque solare de equis.</p>
     <p>Another paragraph. Etiam neque velit, blandit sed scelerisque solare de equis.</p>
     <a href="#">Read more...</a>
</div>
```

If you add class `left` or `right` to the icon, the icon will be taken from the normal flow and placed along the left or right side of its container, and text will wrap around it.

To add an icon to the block you can use all the classes described in chapter **6.5.3 Icons**. Here's an example with less text and bigger icon:

```
<div class="feature">
     <span class="left ic ic-star ic-2x ib ib-size-xl"></span>
     <h2>Block Heading Example</h2>
     <p>This is a paragraph of sample text. Lorem ipsum dolor sit, consectetur adipiscing elit.
Etiam neque velit, blandit sed scelerisque solare de equis. Sibit placem vobetur rios tolbaro
del rei como deus selvince daso rex.</p>
</div>
```

Change color on mouse hover

To change the background color of the iconbox on mouse hover over the entire block, add class `feature-icon-hover` to the main element:

```
<div class="feature feature-icon-hover">
```

```
    <span class="left ic ic-star ib"></span>
    <h2>Block Heading Example</h2>
    <p>This is a paragraph of sample text.</p>
</div>
```

Default background color of the iconbox can be configured in the admin panel.
Refer to **6.5.3.B Icons with background color (iconbox)** for details.

Indented block

To display a block with indentation on the left side, add class `indent` to the main element:

```
<div class="feature feature-icon-hover indent">
    <span class="ic ic-star ib"></span>
    <h2>Block Heading Example</h2>
    <p>This is a paragraph of sample text.</p>
</div>
```

Note that inside indented block, the first level icon automatically floats left so class `left` doesn't need to be added to the icon.

If you increase the iconbox size (by adding a class such as `ib-size-xxl`), you would also need to add class `indent-size-xxl` to the main block. It will adjust the size of the indentation.

```
<div class="feature feature-icon-hover indent indent-size-xxl">
    <span class="ic ic-star ib ib-size-xxl"></span>
    <h2>Block Heading Example</h2>
    <p>This is a paragraph of sample text.</p>
</div>
```

To change the size of the indentation, use the following classes:
`indent-size-l`, `indent-size-xl`, `indent-size-xxl`, `indent-size-xxxl`.

Centered block

To align elements of the block to the center, use class `centered`. Like this:

```
<div class="feature centered">
     <span class="icon">
             <img src="{{media url='wysiwyg/infortis/ultimo/icons/star.png'}}"
                     alt="My description" />
     </span>
     <h3>Title</h3>
     <p>This is sample text</p>
     <p>Another paragraph</p>
</div>
```

6.5.5 Captions for Slideshow

This chapter describes how to create captions inside the homepage Slideshow. Presented code can be used inside the main slides and inside small banners at the side of the Slideshow.

To set background color, use one of the following classes: `light1`, `light2`, `light3`, `dark1`, `dark2`, `dark3`.

For example:

```
<div class="caption light1">
    <p>Lorem ipsum dolor sit amet</p>
</div>
```

To position caption, use the following classes (by default captions is positioned in the bottom left corner): `top`, `top-right`, `bottom-right`.

For example:

```
<div class="caption light1 top-right">
    <h2 class="heading">Big Heading</h2>
    <p>Lorem ipsum dolor sit amet</p>
</div>
```

Additionally, the paragraph of text inside the caption can be aligned right with class: `right`, or centered with class: `centered`.

For example:

```
<div class="caption dark1 centered">
   <h2 class="heading">Big Heading</h2>
   <p>Lorem ipsum dolor sit amet</p>
</div>
```

To stretch the caption, use class: `full-width`. To make it narrower, use: `narrow`.

For example:

```
<div class="caption dark3 full-width">
   <p>Lorem ipsum dolor sit amet dictum sit altum.</p>
</div>
```

On narrow screens elements of the caption will be hidden. To keep selected elements visible, use class: `permanent`.

For example:

```
<div class="caption light1">
    <h2 class="heading permanent">Heading always visible</h2>
    <p>This paragraph will be hidden on mobile</p>
</div>
```

6.6 Theme customization

This chapter was moved to: **5. Customization**

7. Theme settings

This chapter refers to the following section of the theme admin panel:
System > Configuration > Ultimo > Theme Settings

> **Important:**
>
> beginning from this chapter, each chapter refers to a specific sections of theme admin panel. Path of the admin panel section is visible at the top of each page.

7.1 Header

This chapter refers to the following section of the theme admin panel:
System > Configuration > Theme Settings > Header

List of all settings available in that section can be found later in this chapter: **7.1.2 Header configuration options**.

7.1.1 Header structure

Width of the header sections (Left Column, Central Column, Right Column) is specified in grid units. As this is a 12-column grid, each section spans a number of those 12 columns, and should always add up to 12.

Example 1:

If you want to display three equal sections (Left Column, Central Column, Right Column), sum of these three sections has to be equal 12 grid units. For example:

```
Left Column = 4, Central Column = 4, Right Column = 4
4 + 4 + 4 = 12
All the three sections will have equal width.
```

Example 2:

If you want to get rid of one section, e.g. Central Column, leave the **Central Column** field empty and make the Right Column wider. For example:

```
Left Column = 4, Central Column = leave empty, Right Column = 8
4 + ... + 8 = 12
```

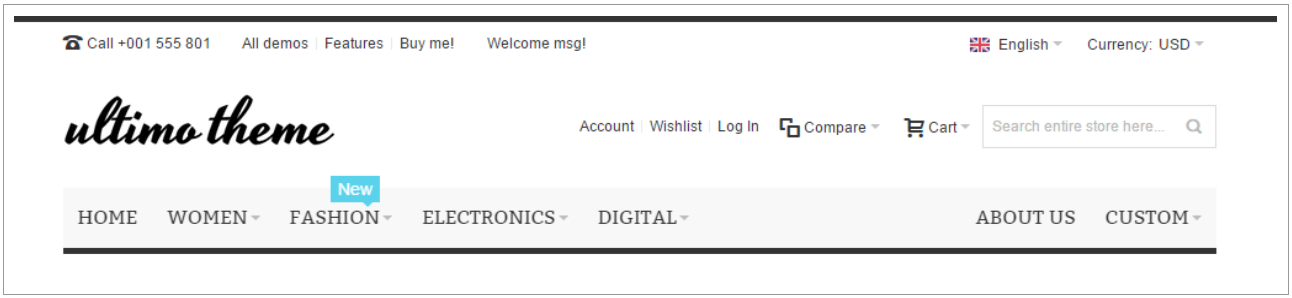
Thanks to that the Central Column will disappear and the Right Column will be 2 times wider than the Left Column:

Now you need to assign the search box to the Right Column instead of the default Central Column. The Central Column is not displayed so any blocks assigned to the Central Column will also not be displayed.

> **Important:**
>
> if given header section is not displayed, then any blocks assigned to that section will also not be displayed.

To assign search box to the Right Column, in the **Position** field under *System > Configuration > Theme Settings > Header > Search* select “Inside User Menu...” and then in the **Position Inside User Menu** field select where the search box should be

displayed. The final effect can look like on the following screenshot:



7.1.2 Header configuration options

In this chapter you can find a list of options available in the following section of the theme admin panel: `System > Configuration > Theme Settings > Header`.

Width of Columns

- **Left Column Width** – width of the left section. Specify value in grid units (between 2 and 12). Important: Leave empty to remove this column.
- **Central Column Width** – width of the central section. Specify value in grid units (between 2 and 12). Important: Leave empty to remove this column.
- **Right Column Width** – width of the right section. Specify value in grid units (between 2 and 12). Important: Leave empty to remove this column.

Logo

- **Position** – select position of the site logo.

Search

- **Position** – select position of the search box. Select `Inside User Menu...` to show another option (see below).
- **Position Inside User Menu** – this field is visible only if `Inside User Menu...` was selected in the field above. Select position of the search box **inside** the User Menu.

User Menu

- Position – select position of the User Menu.
- Position of Additional Line Break – e.g. add a line break after the cart drop-down to force other elements of the User Menu to be displayed in the new line (below the cart). Note that elements of the User Menu are rendered in the following order: cart drop-down, "Compare" block, Top Links, custom static block.
- "Sign Up" Link – if enabled, "Sign Up" link will be displayed next to the "Log In" link in the header.

"Compare" Drop-down

- Display Block "Compare" in the Header – enable/disable "Compare" drop-down block in the header. If set to **No**, the block will be displayed on its default position in Magento: in the left sidebar.
- Show Text Label "Compare" – if set to **No**, label "Compare" will be hidden in the "Compare" drop-down block in the header.

"Cart" Drop-down

- Show Text Label "Cart" – if set to **No**, label "Cart" will be hidden in the cart drop-down block in the header. Note that in order to show cart drop-down block, option "Display Shopping Cart Sidebar" under `System > Configuration > Checkout > Shopping Cart Sidebar` has to be enabled.

General

- Mobile Header – transform the standard header into a mobile-friendly header on small mobile devices (if browser viewport is narrower than 770 pixels).

> **Important:**
>
> if mobile header is enabled, option `System > Configuration > Menu > Main Menu Mode` has to be set to **Drop-down/Mobile** or **Mobile**.

- Move Switchers Next To Logo – if set to **Yes**, if mobile header is enabled, language and currency switchers will be moved next to the logo (but only in mobile header) to make the header smaller.

- **Sticky Header** – if enabled, the main menu does not disappear when the user scrolls down the page.

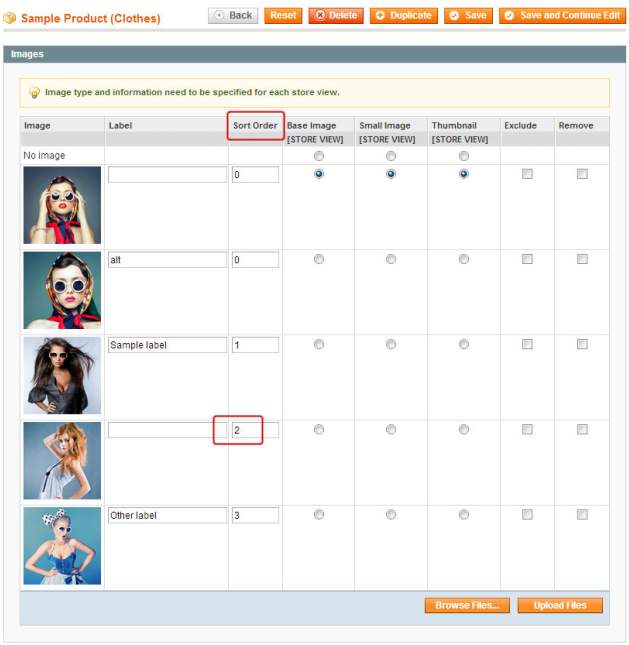
7.2 Category View

In this tab, you can configure the options related to the category view. Category view presents the list of products from the selected category.

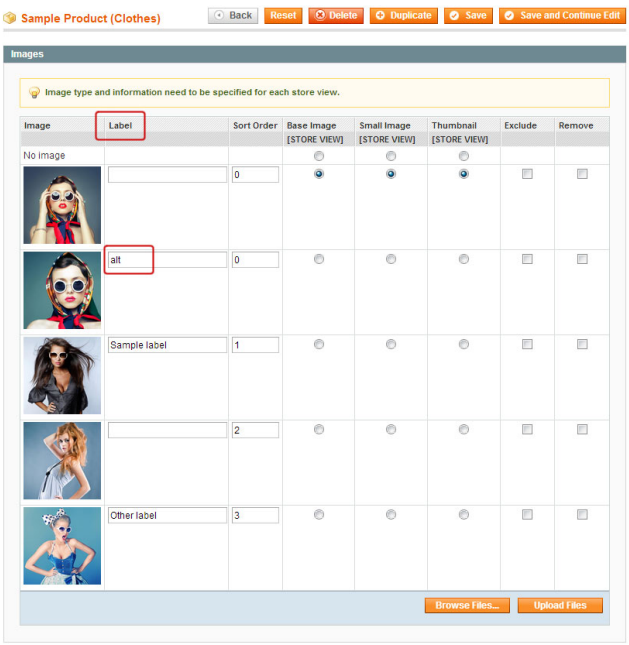
Images

- **Keep Image Aspect Ratio** – set **Yes**, to keep the aspect ratio of the product images intact. Height of the images will be calculated automatically based on width. This option also affects product listings: **Featured Products Slider** block and **New Products** block.
- **Alternative Image** – show alternative image on mouse hover over the main image.
- **Select Alternative Image By Column** – which column (attribute) of the product image gallery will mark the alternative image. Available columns:
  - Label
  - Sort Order

  (See examples on the next page.)
- **Alternative Image Column Value** – specify the value of the column (attribute) which will mark alternative images of your products (see examples on screenshots below).



Example 1



Example 2

Example 1:

if **Select Alt. Image By Column** is set to `Sort Order`, specify the number which will mark alternative images, e.g. `2`.

> **Note:**
>
> in this example `2` doesn't mean second image, but the image which has value `2` in the **Sort Order** column in the image gallery.

Example 2:

if **Select Alt. Image By Column** is set to `Label`, specify label which will mark alternative images of your products. The label should be a short single word, e.g. `alt`.

7.3 Category View (Grid Mode)

In this tab, you can configure the options related to the **grid mode** of the category view.

Number of Columns

- **Number of Columns** – number of products displayed in a single row, if the **Maximum Page Width** is set to 1024 pixels (which is the lowest predefined value of the page width).

  If the **Maximum Page Width** is larger, the number of columns will be automatically increased to show more products in a row. For example: if the **Number of Columns** = `3` and the **Maximum Page Width** = `1280 px`, on wide screens the grid will display 4 products in a row.

- **Number of Columns Below 768px** – number of columns displayed if browser viewport width is between 640 px and 768 px.

- **Number of Columns Below 640px** – number of columns displayed if browser viewport width is between 480px and 640 px.

- **Number of Columns Below 480px** – number of columns displayed if browser viewport width is between 320 px and 480 px. Below that width products are displayed in a single column.

Display

- **Equal Height of Grid Items (Products)** – if enabled, all grid items will have the same height and "Add to cart" button will be displayed at the bottom of the item. This option is limited to product grid in category view and will not be applied to product sliders.

- **Product Hover Effect** – enable/disable effect which is visible on mouse hover over the product: block is emphasized with frame and shadow.

> **Note**:
>
> enable in order to use option "Display On Hover" in **Display Selected Elements**. Otherwise displaying elements on mouse hover over the product

> will not work properly.

- **Disable Hover Effect Below...** – product Hover Effect will be disabled if width of the browser viewport is below selected value.
- **Hide "Add to" Links Below...** – "Add to wishlist" and "Add to compare" links will be hidden if width of the browser viewport is below selected value.
- **Align Center** – align to the center elements of the grid item: product name, price, button etc.

Display Selected Elements

In this section, you can configure the display of selected elements of the grid items (products). Each of the elements can be: **a)** disabled and not displayed in the grid, **b)** displayed in a standard way, **c)** displayed only on mouse hover over the product.

Below is the list of the elements which can be configured:

- **Name** – name of the product
- **Display Name in Single Line** – If set to "Yes": if product name is too long to be displayed in a single line, it will be clipped and an ellipsis (…, U+2026 HORIZONTAL ELLIPSIS) will be appended at the end of the line to represent the clipped text. This will only take effect if the field **Name** is set to "Display".
- **Price** – price of the product
- **Ratings** – product ratings (stars and number of reviews)
- **Add To Cart** – "Add to cart" button
- **"Add to" Links** – "Add to wishlist" and "Add to compare" links
- **Show "Add to" Links as Icons Above the Image** – if set to `No`, "Add to wishlist" and "Add to compare" links will be displayed as text links. Otherwise, links will be displayed as simple icons above the product image.

> **Note**:
>
> this option also affects "Add to wishlist" and "Add to compare" links in product sliders: **Featured Products Slider** and **New Products Slider**.

7.4 Category View (List Mode)

In this tab, you can configure the options related to the **list mode** of the category view.

- **Product Hover Effect** – enable/disable effect which is visible on mouse hover over the product: block is emphasized with frame and shadow.
- **Show "Add to" Links as Text Links** – If set to **No**, "Add to wishlist" and "Add to compare" links will be displayed as text links with icons.

7.5 Product Page

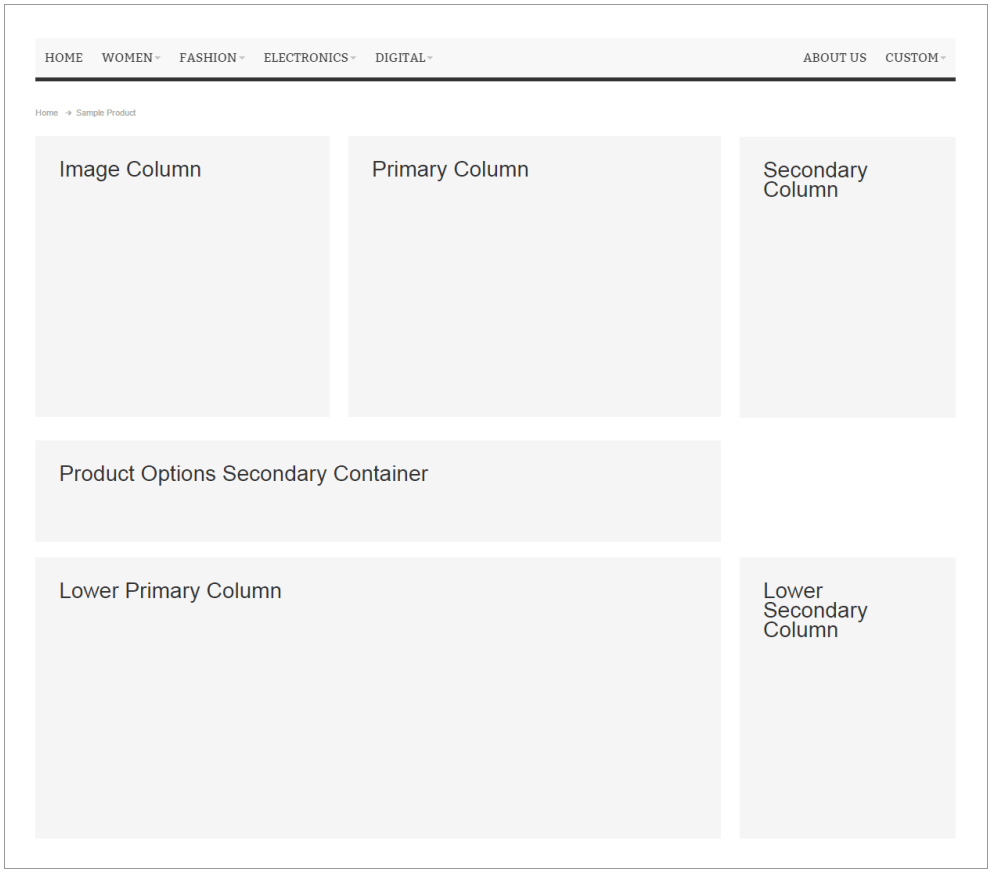
This chapter refers to the following section of the theme admin panel:
System > Configuration > Theme Settings > Product Page

List of all settings available in that section can be found later in this chapter: **7.5.3 Product page configuration options**.

7.5.1 Product page structure

Width of the product page sections is specified in grid units. As this is a 12-column grid, each section spans a number of those 12 columns, and should always add up to 12 for each row.

Below you can see all available sections:

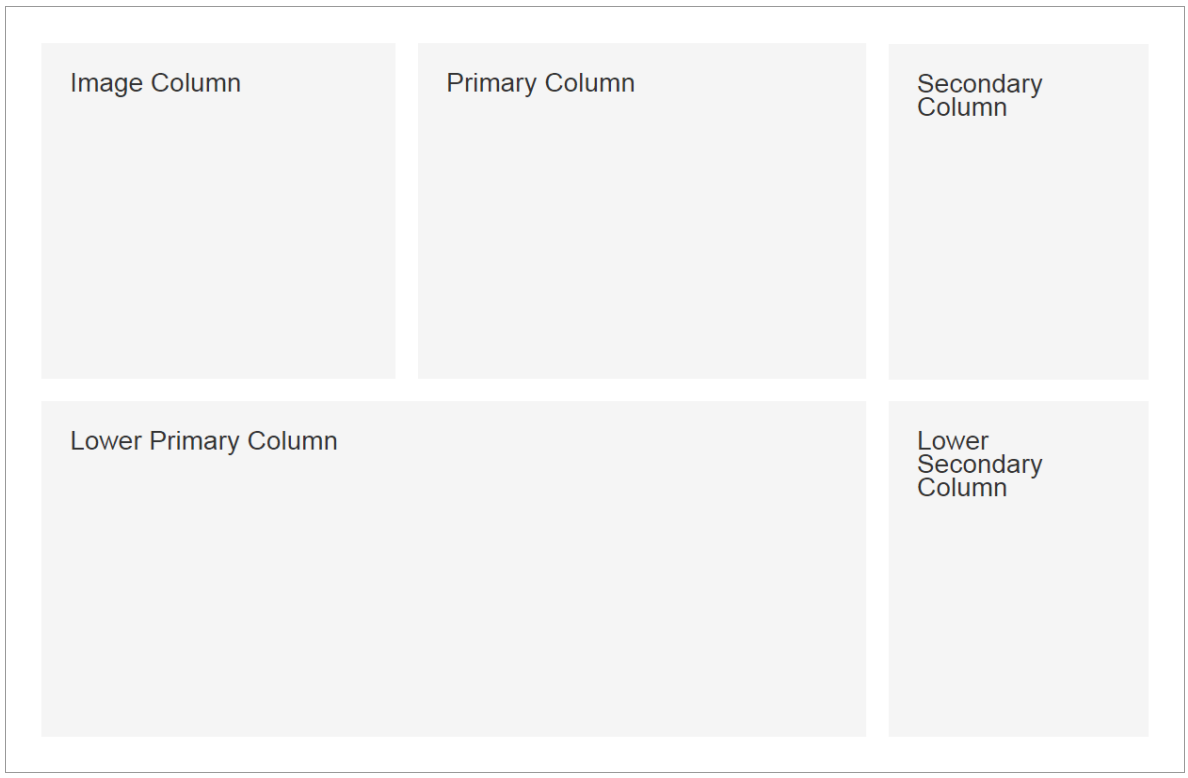


Example 1:

If you want to display three main sections (Image Column, Primary Column, Secondary Column) in a single row, sum of these three sections has to be equal **12 grid units**. For example:

```
Image Column = 4, Primary Column = 5, Secondary Column = 3
4 + 5 + 3 = 12
All the main three sections will be displayed in one row.

Lower Primary Column = 9, Lower Secondary Column = 3
9 + 3 = 12
Lower sections will be displayed in another row, below the main sections.
```

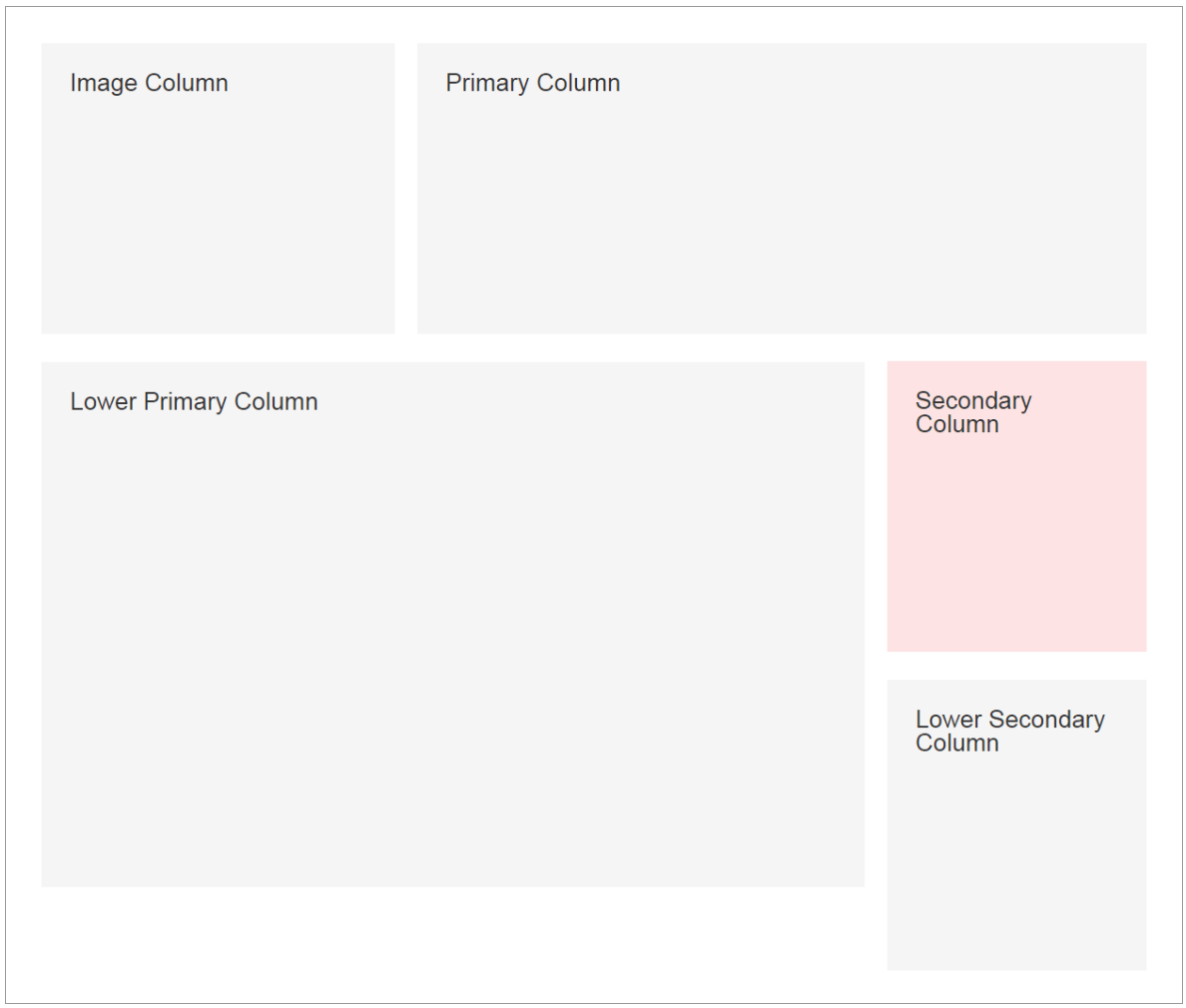
Note that the Product Options Secondary Container is optional (see more info later in this chapter) and will be displayed only if product options were assigned to "Block after Info Column" in product's settings. Otherwise, options container will not be displayed:



But if sum of these three sections is greater than 12, the last section will drop to the next row. For example:

```
Image Column = 4, Primary Column = 8, Secondary Column = 3
4 + 8 + 3 = 15
Image Column and Primary Column will be displayed in one row, and Secondary Column will drop to
the next row.

Lower Primary Column = 9, Lower Secondary Column = 3
9 + 3 = 12
```

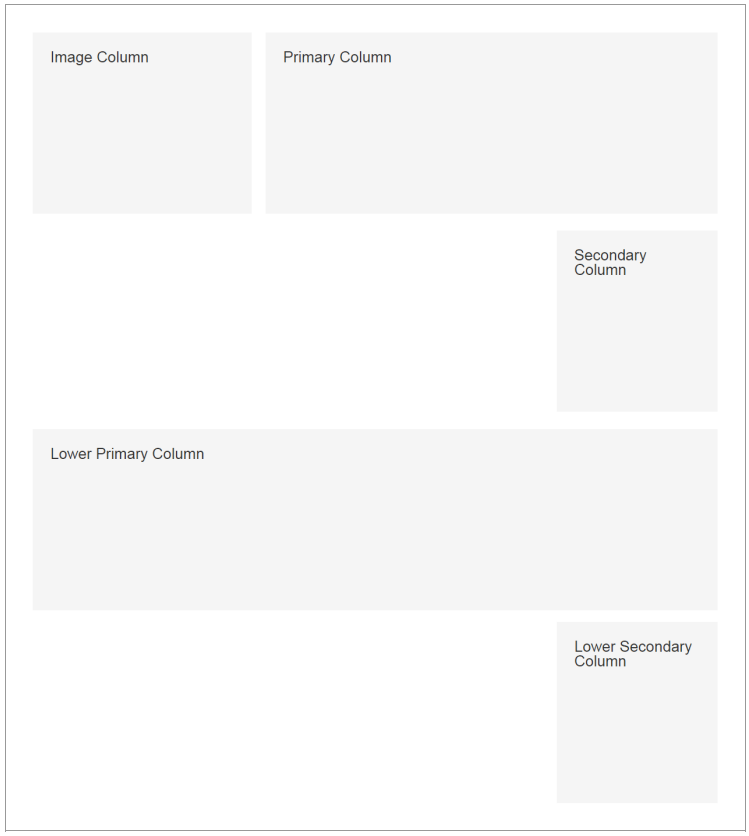
Note that the Secondary Column always **floats to the right side** so it will be placed above the Lower Secondary Column and at the side of the Lower Primary Column:



If the Lower Primary Column is wider, then it will be displayed in another row, below the Secondary Column. For example:

```
Image Column = 4, Primary Column = 8, Secondary Column = 3
4 + 8 + 3 = 15
Secondary Column will drop to the next row.

Lower Primary Column = 12, Lower Secondary Column = 3
12 + 3 = 15
```

The Lower Primary Column takes entire available space so it will drop to the next row, below the Secondary Column. And the Lower Secondary Column will drop to another row, below the Lower Primary Column.



In this case it's recommended to make the Lower Primary Column narrower so that it fits the empty space at the left side of the Secondary Column.

Example 2:

If you want to get rid of the Secondary Column, leave the **Secondary Column Width** field empty and make the Primary Column wider. For example:

```
Image Column = 4, Primary Column = 8, Secondary Column = leave empty
4 + 8 + ... = 12

Lower Primary Column = 9, Lower Secondary Column = 3
9 + 3 = 12
```

The Image Column and the Primary Column will take the entire row and the Secondary Column will disappear:

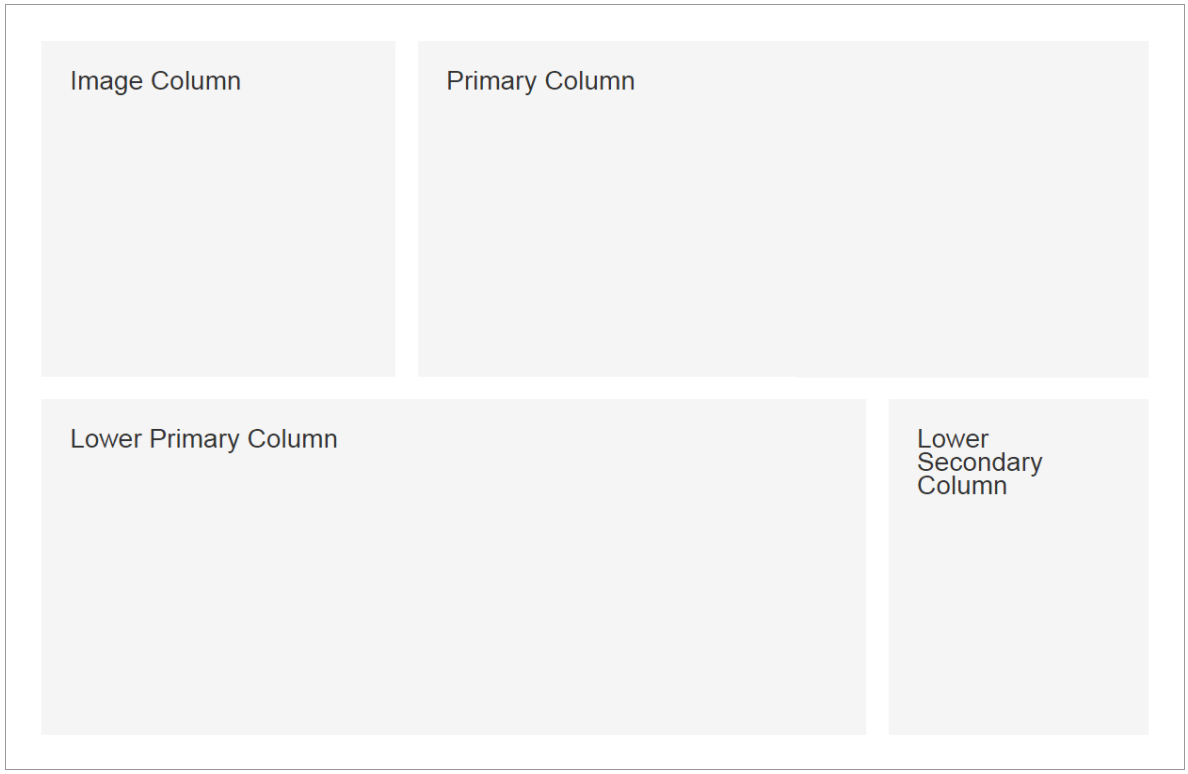


Image Column
Primary Column
Lower Primary Column
Lower Secondary Column

Example 3:

If you want to get rid of both: the Secondary Column and the Lower Secondary Column, leave the **Secondary Column Width** and the **Lower Secondary Column Width** fields empty and make the Primary Column and the Lower Primary Column wider. For example:

```
Image Column = 4, Primary Column = 8, Secondary Column = leave empty
4 + 8 + ... = 12

Lower Primary Column = 12, Lower Secondary Column = leave empty
12 + ... = 12
```

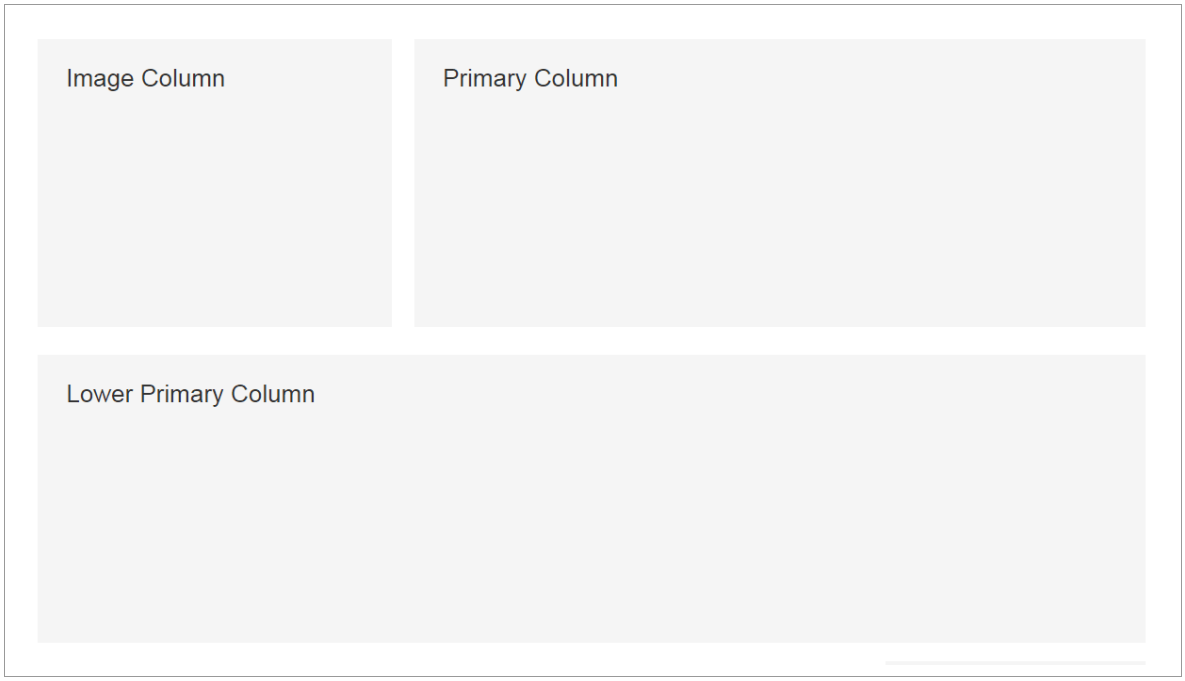


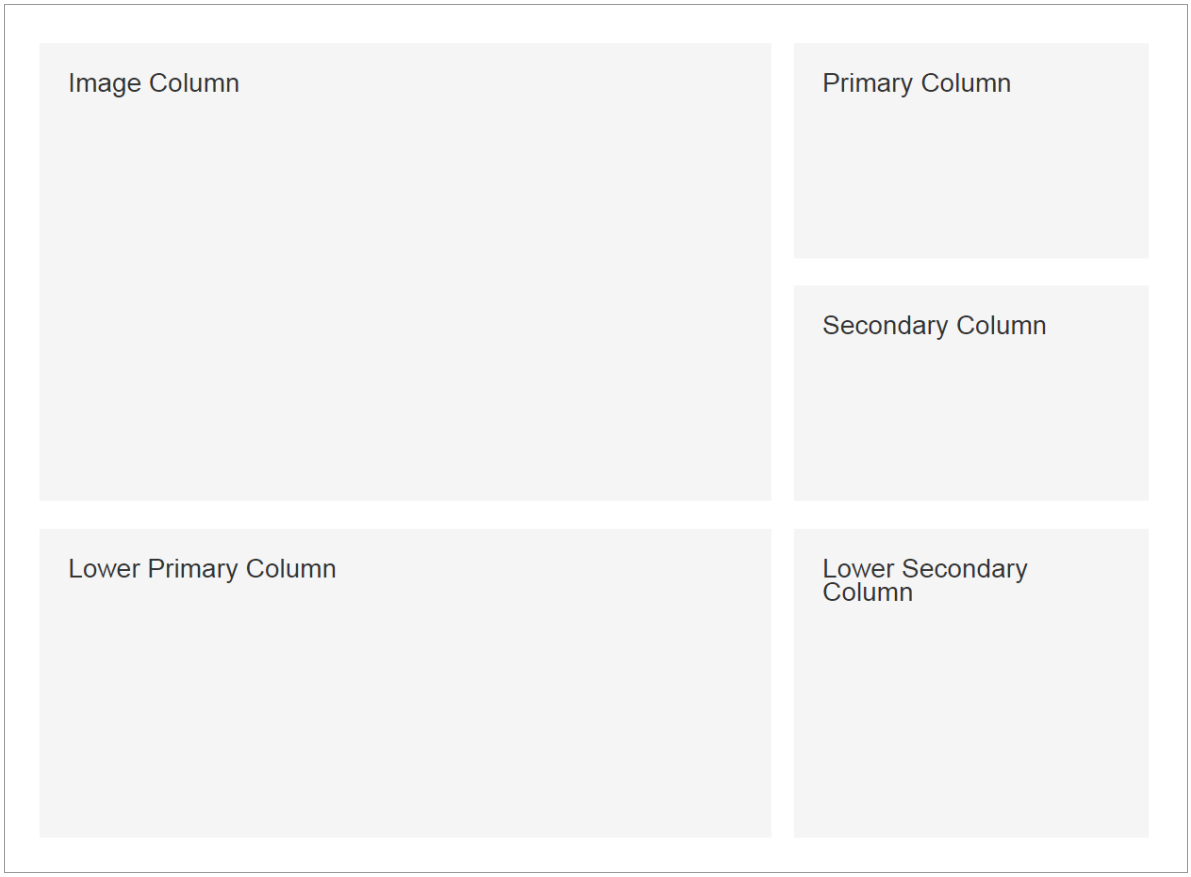
Image Column
Primary Column
Lower Primary Column

Example 4:

If sum of the Image Column and the Primary Column equals 12, and the Primary Column and the Secondary Column both have the same width, sections will have the following layout:

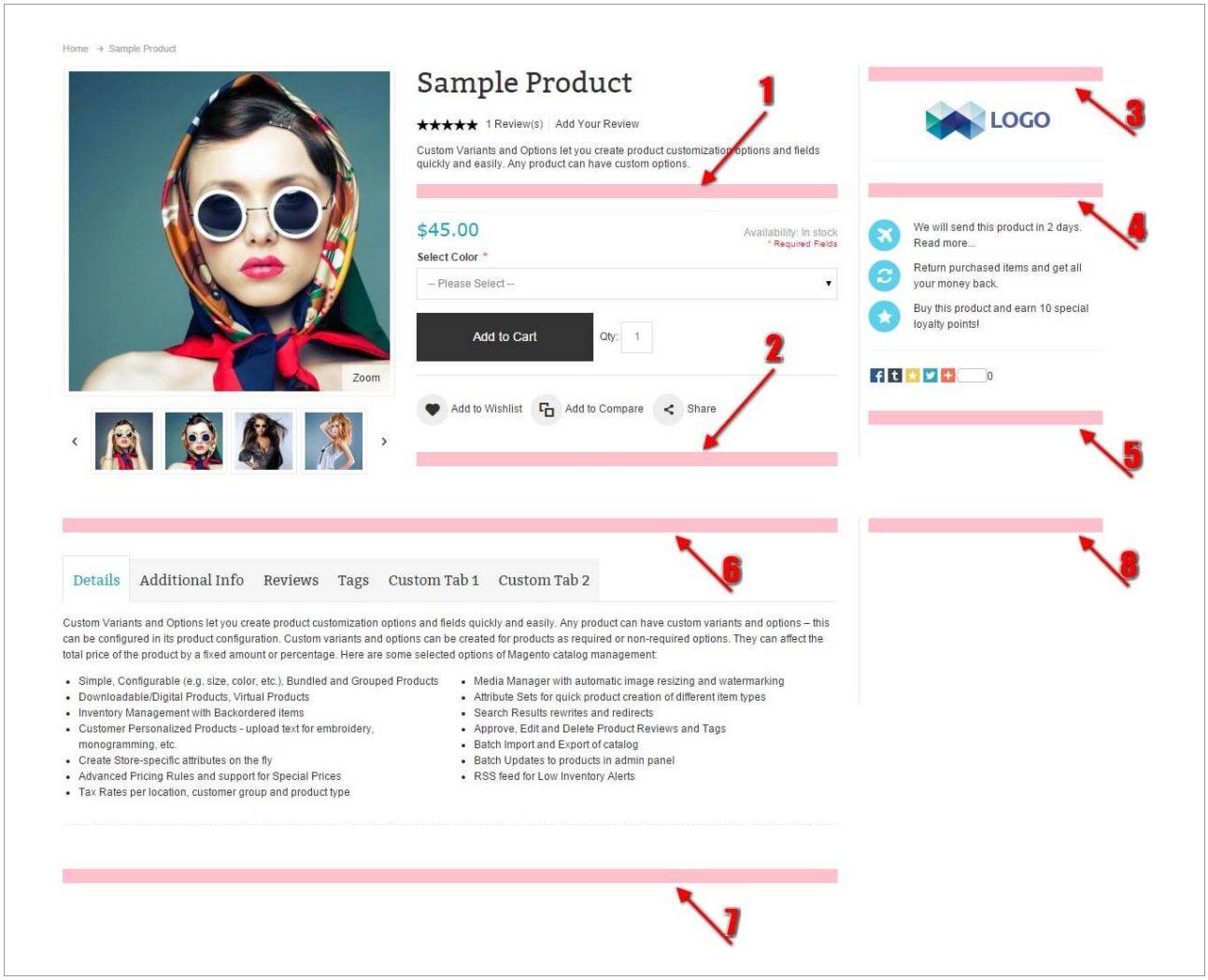
```
Image Column = 8, Primary Column = 4, Secondary Column = 4
8 + 4 + 4 = 16

Lower Primary Column = 9, Lower Secondary Column = 3
8 + 4 = 12
```

This allows you to present large product images. In this example, both the Primary Column and the Secondary Column will be displayed as a simple sidebar:



7.5.2 Product page block positions

In section `System > Configuration > Theme Settings > Product Page` of admin panel you can select where to display some of the product page blocks (i.e. select positions of those blocks). Below you can see a list of all available positions:


Home → Sample Product
Sample Product
1 Review(s) | Add Your Review
Custom Variants and Options let you create product customization options and fields quickly and easily. Any product can have custom options.
$45.00
Availability: In stock
* Required Fields
Select Color *
-- Please Select --
Add to Cart
Qty: 1
Add to Wishlist
Add to Compare
Share
Zoom
LOGO
We will send this product in 2 days. Read more...
Return purchased items and get all your money back.
Buy this product and earn 10 special loyalty points!
Details
Additional Info
Reviews
Tags
Custom Tab 1
Custom Tab 2
Custom Variants and Options let you create product customization options and fields quickly and easily. Any product can have custom variants and options – this can be configured in its product configuration. Custom variants and options can be created for products as required or non-required options. They can affect the total price of the product by a fixed amount or percentage. Here are some selected options of Magento catalog management:
Simple, Configurable (e.g. size, color, etc.), Bundled and Grouped Products
Downloadable/Digital Products, Virtual Products
Inventory Management with Backordered items
Customer Personalized Products - upload text for embroidery, monogramming, etc.
Create Store-specific attributes on the fly
Advanced Pricing Rules and support for Special Prices
Tax Rates per location, customer group and product type
Media Manager with automatic image resizing and watermarking
Attribute Sets for quick product creation of different item types
Search Results rewrites and redirects
Approve, Edit and Delete Product Reviews and Tags
Batch Import and Export of catalog
Batch Updates to products in admin panel
RSS feed for Low Inventory Alerts
1
2
3
4
5
6
7
8

1. Primary Column, Position 1
2. Primary Column, Position 2
3. Secondary Column, Position 1

4. Secondary Column, Position 2
5. Secondary Column, Position 3
6. Lower Primary Column, Position 1
7. Lower Primary Column, Position 2
8. Lower Secondary Column, Position 1

7.5.3 Product page configuration options

In this chapter you can find a list of options available in the following section of the theme admin panel: `System > Configuration > Theme Settings > Product Page`.

Width of Columns

- Image Column Width – width of the section with the product image. Specify value in grid units. By default it is equal to 4.

  If you change the value of the **Image Column Width**, you should also change the width of the main product image. You can do it in `System > Configuration > Zoom > Image Size > Main Image Width`. Optimal width of the image depends on two values:

  - the width specified in the **Image Column Width** field
  - the maximum width of the page specified in **Maximum Page Width** field in `System > Configuration > Ultimo > Theme Layout` section of the admin panel.

  For example, if your **Maximum Page Width** is equal to 1024 px, and the **Image Column Width** is 4 units, there is no need to set the image width larger than 287 pixels, because the visible size of the image displayed inside the Image Column will not exceed the size of the Image Column.

  The following table specifies the optimal **Main Image Width** for different sizes of the Image Column:

	Maximum Page Width				
Image Column Width	**1024 px**	**1280 px**	**1360 px**	**1440 px**	**1680 px**
4 units	287	363	395	421	466
5 units	366	463	504	536	593
6 units	446	563	612	651	720

- **Primary Column Width** – width of the section with the product name, price, options etc. Specify value in grid units. By default it is equal to 5.
- **Secondary Column Width** – width of the section with additional product data and custom content. Specify value in grid units. By default it is equal to 3.

> **Important:**
> leave empty to remove this column.

- **Lower Primary Column Width** – width of the section with tabs, reviews etc. Specify value in grid units. By default it is equal to 9.
- **Lower Secondary Column Width** – width of the section with additional custom content. Specify value in grid units. By default it is equal to 3.

> **Important:**
> leave empty to remove this column.

- **Product Options Secondary Container Width** – width of the section with product options container. Specify value in grid units. By default it is equal to 9. It's recommended to specify the same width as in the **Lower Primary Column Width** field.

  Note that product options will be displayed in this section only if:

  - product has any option
  - the "Display Product Options In" field in product's configuration (under `Catalog > Manage Products`) was set to "Block after Info Column".

  If the "Display Product Options In" field was set to "Product Info Column", product options will be displayed inside the Primary Column.

Collateral Data

Container for product collateral data. It can be displayed as tabs, accordion or stacked blocks.

- Display Collateral Data as Tabs – display Collateral Data as tabs or accordion. If set to "No", Collateral Data will be displayed as stacked blocks.
- Mode – available modes of the tabs:
  - **Tabs** – display Collateral Data in tabs.
  - **Tabs/Accordion** – display Collateral Data in tabs which turn into accordion if browser viewport is narrower than the width specified in the "Accordion Threshold" field.
  - **Accordion** – display Collateral Data in accordion.
- Accordion Threshold – if option "Mode" was set to "Tabs/Accordion", tabs will turn into accordion if browser viewport is narrower than the width specified in this field. Enter value in pixels. This option is available only if "Mode" was set to "Tabs/Accordion".
- Accordion Initially Collapsed – if set to "Yes", if accordion is visible on page load, all blocks of the accordion will be collapsed. This option is available only if accordion was enabled.
- Small Size – display small tabs/accordion. Useful when Collateral Data is displayed in a narrow section of the product page.
- Position – position of Collateral Data container. Screenshot with all available positions can be found in this chapter: **7.5.2 Product page block positions**.

Collateral Data - Selected Blocks

Select blocks which will be displayed in the Collateral Data container. Collateral Data can be displayed in different modes: tabs, accordion or stacked blocks. For more info, see the section above.

- Reviews in Collateral Data – display product reviews in Collateral Data container.
- Tags in Collateral Data – display product tags in Collateral Data container.
- Related Products in Collateral Data – display related products in Collateral Data container.

- Up-sell Products in Collateral Data – display up-sell products in Collateral Data container.

Related Products

- Position – position of the Related Products block on product page. Screenshot with all available positions can be found in this chapter: **7.5.2 Product page block positions**.

> **Note:**
>
> this option **will be ignored** if option "Related Products in Collateral Data" was set to "Yes".

- Related Products Layout – layout of the Related Products block on product page:
  - **Simple Slider** – standard horizontal slider.
  - **Thumbnails Slider** – in this case you can specify (in the "Number of Visible Products" field) the number of products displayed in a single slide.
- Breakpoints – with this parameter you can specify the number of products in a row visible with a particular browser viewport width.

> **Note:**
>
> this option is available only if the "Related Products Layout" field was set to "Simple Slider".

Use the format presented in the examples below. Leave this field empty to use the default value.

Value of this parameter consists of pairs of numbers in brackets separated by comma. In each pair of numbers `[A, B]` number `A` indicates the web browser viewport width, number `B` indicates the number of products visible if browser viewport width is greater that the width specified in `A`.
In other words: display `B` products if viewport width is above `A` pixels.

- **Example 1 – the default value:**

```
[0, 1], [320, 2], [480, 3], [768, 1]
```

The default value is useful if the slider is displayed inside narrow sections, for example, inside the Secondary Column or Lower Secondary Column.

The default value can be translated as: display 1 product if viewport width is between 0px and 320px, display 2 products if viewport width is above 320px, display 3 products if viewport width is above 480px, and display 1 product if viewport width is above 768px.

This means that the wider is the screen, the more products will be visible in the slider, but **if the screen is wider than 768px** – only 1 product will be visible. Thanks to that, the slider can be displayed inside narrow columns.

- **Example 2:**

```
[0, 1], [320, 2], [480, 3], [960, 4], [1280, 5]
```

This example is useful if the slider is displayed inside wide section, for example: inside the Lower Primary Column.

This example can be translated as: display 1 product if viewport width is between 0px and 320px, display 2 products if viewport width is above 320px, display 3 products if viewport width is above 480px, display 4 products if viewport width is above 960px, and display 5 products if viewport width is above 1280px.

- **Example 3:**

```
[0, 3]
```

which can be translated as: always display 3 products.

- Number of Visible Products – maximum number of related products displayed in a single slide. This field is visible only if "Thumbnails Slider" was selected in the "Related Products Layout" field.

- Timeout – to enable automatic scrolling of the slider, specify time (in milliseconds) between transitions. Leave empty to disable automatic scrolling. Note that 1000 miliseconds = 1 second.

- Replace Related Products With Static Block – **this option is deprecated and it will be removed**. Related Products can be replaced with a static block. To replace Related Products, first create a static block with ID

`block_product_replace_related`.

> **Note:**
>
> this option **will be ignored** if option "Related Products in Collateral Data" was set to "Yes".

You can choose one of the following options:

- **Disable Completely** – disable Related Products block and don't replace it with any other content.
- **Don't Replace With Static Block** – never replace Related Products block with custom static block.
- **If Empty, Replace With Static Block** – replace Related Products block with custom static block only if the product does not have any related products.
- **Replace With Static Block** – always replace Related Products block with custom static block. It's useful when you want to disable Related Products block but still want to display some content in the same area.

Up-sell Products

- Position – position of Up-sell Products block on product page. Screenshot with all available positions can be found in this chapter: **7.5.2 Product page block positions**.

> **Note:**
>
> this option **will be ignored** if option "Up-sell Products in Collateral Data" was set to "Yes".

- Breakpoints – with this parameter you can specify the number of products in a row visible with a particular browser viewport width.

  Use the format presented in the examples below. Leave this field empty to use the default value.

  Value of this parameter consists of pairs of numbers in brackets separated by comma. In each pair of numbers `[A, B]` number `A` indicates the web browser viewport width, number `B` indicates the number of products visible if browser

viewport width is greater that the width specified in `A`.
In other words: display `B` products if viewport width is above `A` pixels.

- **Example 1 – the default value:**

```
[0, 1], [320, 2], [480, 3], [960, 4], [1280, 5]
```

The default value is useful if the slider is displayed inside wide section, for example: inside the Lower Primary Column.

The default value can be translated as: display 1 product if viewport width is between 0px and 320px, display 2 products if viewport width is above 320px, display 3 products if viewport width is above 480px, display 4 products if viewport width is above 960px, and display 5 products if viewport width is above 1280px.

- **Example 2:**

```
[0, 1], [320, 2], [480, 3], [768, 1]
```

This example is useful if the slider is displayed inside narrow sections, for example, inside the Secondary Column or Lower Secondary Column.

This example can be translated as: display 1 product if viewport width is between 0px and 320px, display 2 products if viewport width is above 320px, display 3 products if viewport width is above 480px, and display 1 product if viewport width is above 768px.

This means that the wider is the screen, the more products will be visible in the slider, but **if the screen is wider than 768px** – only 1 product will be visible. Thanks to that, the slider can be displayed inside narrow columns.

- **Example 3:**

```
[0, 3]
```

which can be translated as: always display 3 products.

- Timeout – to enable automatic scrolling of the slider, specify time (in milliseconds) between transitions. Leave empty to disable automatic scrolling. Note that 1000 miliseconds = 1 second.

- Replace Up-sell Products With Static Block – **this option is deprecated and it will be removed**. Up-sell Products can be replaced with a static block. To replace

Up-sell Products, first create a static block with ID `block_product_replace_upsell`.

> **Note:**
>
> this option **will be ignored** if option "Up-sell Products in Collateral Data" was set to "Yes".

You can choose one of the following options:

- **Disable Completely** – disable Up-sell Products block and don't replace it with any other content.
- **Don't Replace With Static Block** – never replace Up-sell Products block with custom static block.
- **If Empty, Replace With Static Block** – replace Up-sell Products block with custom static block only if the product does not have any up-sell products.
- **Replace With Static Block** – always replace Up-sell Products block with custom static block. It's useful when you want to disable Up-sell Products block but still want to display some content in the same area.

Other Elements

- Show SKU – display SKU on product page. SKU is an unique identifier of a product.

7.6 Product Labels

In this tab, you can configure the options related to product labels.

> **Important:**
>
> in order to display product labels, you may need to reindex the data (**4.2 Index management**) and flush the cache (**4.1 Caching system**) after marking products as new and after setting a special price.

- Show "New" Label – if set to **Yes**, will show a "New" label on the main image of all new products. The label will be displayed in both product and catalog views.

> **Note:**
>
> To promote any product as new, go to `Catalog > Manage Products` and select the product. In **Set Product as New from Date** and **Set Product as New from Date** fields, enter the date range in which the product will be promoted as a new product.

- Show "Sale" Label – if set to **Yes**, will show a "Sale" label on the main image of all products with special price. The label will be displayed in both product and catalog views.

> **Note:**
>
> To set a special price go to `Catalog > Manage Products`, select the product and enter a special price in **Special Price** field.
>
> In **Special Price From/To Date** field you can enter a date range within which the special price will be active. This price will apply automatically when the **From Date** arrives and deactivate automatically when the **To Date** arrives. If

> you leave this range blank, the **Special Price** will apply until you remove it.

7.7 Footer

In this tab, you can configure the options related to the footer section.

- **Automatic Column Width Calculation** – if enabled, width of the columns in the footer (two rows of static blocks) will be calculated automatically and columns will always have maximum available width – grid units will be applied to columns. Refer to **6.4 Grid system** for more details about grid units and grid system.

  It will take effect on 2 rows of static blocks with the following identifiers:

  - **first row:**
    - block_footer_column1
    - block_footer_column2
    - …
    - block_footer_column6
  - **second row:**
    - block_footer_row2_column1
    - block_footer_row2_column2
    - …
    - block_footer_row2_column6

  For example, if only 3 static blocks are enabled in the first row, those 3 blocks will be displayed as 3 equal columns.

  If this option is disabled, then the width of each column will be equal to 2 grid units.

> **Note:** this option will not be applied if exactly 5 blocks are enabled. This is due to limitations of the 12-column grid system – it isn't possible to display 5 equal columns, because 12 can't be divided by 5.

7.8 Product sliders

In this tab, you can configure the options related to product sliders.

Settings from this tab will be applied to all existing product sliders, also to Related Products and Ups-Sell Products on product page.

> **Note:**
>
> ***Timeout*** for *Related Products* and *Ups-Sell Products* can be specified individually, refer to **7.5 Product Page** for more details.

More info about product sliders can be found in **6.1.4 Featured Products Slider** and **6.1.5 New Products Slider**.

- **Timeout** – to enable automatic scrolling of the product sliders, you need to specify the time (in milliseconds) between slide transitions. Leave it empty to disable automatic scrolling. Note that 1000 milliseconds = 1 second.
- **Speed** – duration (in milliseconds) of scrolling animation. Applied to scrolling with arrows (and touch swipe).
- **Automatic Scrolling Speed** – duration (in milliseconds) of scrolling animation. Applied to automatic scrolling (if **Timeout** was specified) and to scrolling with pagination.
- **Pause on Hover** – if enabled, when the slider is mouseovered then the automatic scrolling will pause.
- **Loop** – if enabled, scrolling will loop.
- **Lazy Loading** – delays loading of images. If enabled, images outside of viewport will not be loaded before user scrolls to them.

7.9 Default Magento blocks

In this tab, you can disable some of the default Magento blocks. If needed, some of these blocks can be then substituted by a corresponding static blocks. Refer to **6.2 Built-in CMS blocks** for more details about the static blocks provided with this theme.

- **Top Links** – links in the header section: Account, Wishlist, Log In. This block can be substituted by two static blocks: `block_header_top_links` and `block_header_top_links2`.

- **Footer Links** – links in the footer section: Site Map, Search Terms, Contact Us etc. This block can be substituted by the static block `block_footer_links`.

- **Newsletter Subscribe Form** – block with the newsletter subscription form located in the footer section. All visitors have the option to sign up by entering their email address. This block can be substituted by the static block `block_footer_primary_bottom_right`.

  Enter the following code in the block content to display newsletter subscription form inside that block:

```
{{block type="newsletter/subscribe" template="newsletter/subscribe.phtml"}}
```

  By using this code, you can display newsletter subscription form in any other static block or on custom CMS page.

- **Store Switcher in the Footer** – store switcher displayed at the bottom of the page. It is only displayed if there is more than one active store in Magento. Note that this is something different than store view switcher. “Store” in this case is a group of store views. Refer to this article for more details:

  How Multiple Websites & Stores Work

- **Product Page: Related Products Checkbox** – checkboxes displayed in the Related Products block (for products without additional options) on product page.

7.10 Disable Sidebar Blocks (On Home Page)

In this tab, you can remove some of Magento's default blocks **from the home page** sidebars.

7.11 Customization Settings

Settings in this tab are related to theme customization. For more information about theme customization refer to chapter **5.1 How to modify the theme**.

- **Load Custom CSS** – if enabled, `custom.css` file (in which you can override the default styling of the theme) will be loaded in the frontend.

  This might be useful when you want to make some changes in the theme design. All the CSS placed in this file **will not be lost** after upgrading the theme.

> **Note**:
>
> this option may be enabled/disabled individually for each store view.

To create `custom.css` you can use the file `SAMPLE.custom.css` which is located in `skin/frontend/ultimo/default/css/` directory. Simply create a copy of `SAMPLE.custom.css` and rename it to `custom.css`. Uncomment selected code or add your own to override the default styles of the theme.

In some situations custom CSS is not enough. When you need to modify theme's default template files, you should create a custom sub-theme. Refer to **5. Customization** for more details about custom sub-themes in Magento.

7.12 Data Import

This tab is responsible for the data import procedures after installation.

Theme Configuration Import/Export

To import/export theme configuration, go to the **Data Import** tab under `System > Configuration > Ultimo > Theme Settings` and click the appropriate button to display a form with additional settings:

Both options are described below.

- ## Import

  – click this button to display data import form. Using this form you can quickly import theme configuration settings from any of the demo sites. It will import all the theme settings which can be found in sections:
  System > Configuration > Ultimo and
  System > Configuration > Infortis Extensions.

  After importing you can go to those sections and adjust the configuration to your needs. You can import as many times as you want, so import function can be used to reset the configuration to default values. **To import, follow the instructions below:**

Import Configuration
Reset Form
Import Configuration
Import settings
Select Configuration to Import *
Demo2
Configuration Scope *
Default Config
Imported configuration settings will be applied to selected scope (selected store view or website). If you're not sure what is 'scope' in Magento system configuration, it is highly recommended to leave the default scope 'Default Config'. In this case imported configuration will be applied to all existing store views.

1. On the form, choose one of the available demos from the **Select Configuration to Import** list.

2. From the **Configuration Scope** list select configuration scope – imported configuration settings will be saved in the selected scope.

> **Note:**
>
> it is recommended to import to the “Default Config” scope. Especially if you have only one store view in your Magento or if want all your store views to look similar. Settings from the “Default Config” scope can be later overridden on store view level, if needed.

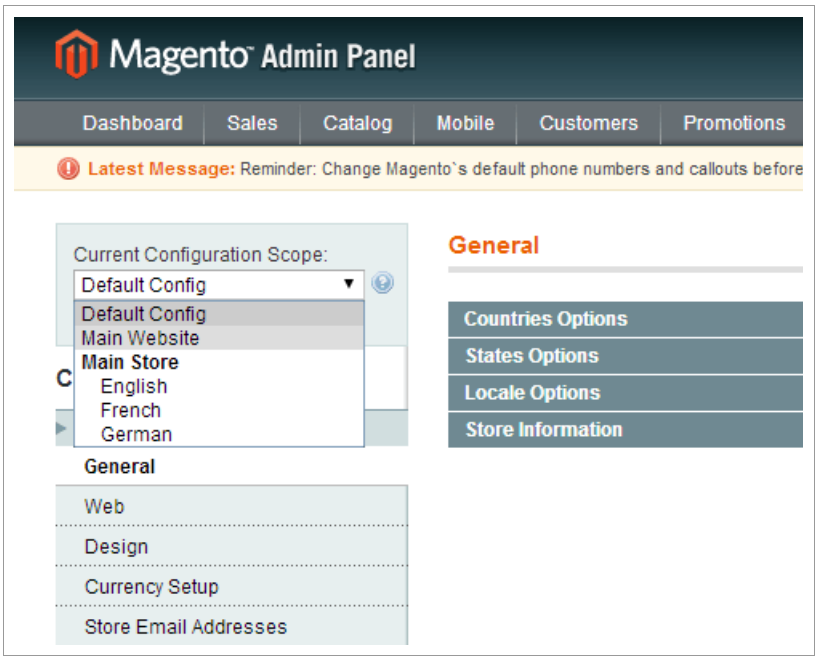
> **Important:**
>
> note that the scope of Magento configuration has three levels:

> ***1.** "Default Config", **2.** Website, **3.** Store view*
>
> Settings of a store view override settings of a website. And settings of a website override settings of the "Default Config".

3. If you import to the `Default Config` scope, imported settings will be applied to all existing store views in your Magento. If you're not sure what is “scope” in Magento system configuration, please refer to the following page: *http://www.magentocommerce.com/knowledge-base/entry/understanding-store-scopes*

4. Finally, click the **Import Configuration** button. After that, you may also need to flush Magento cache (**4.1 Caching system**) in order to see changes.

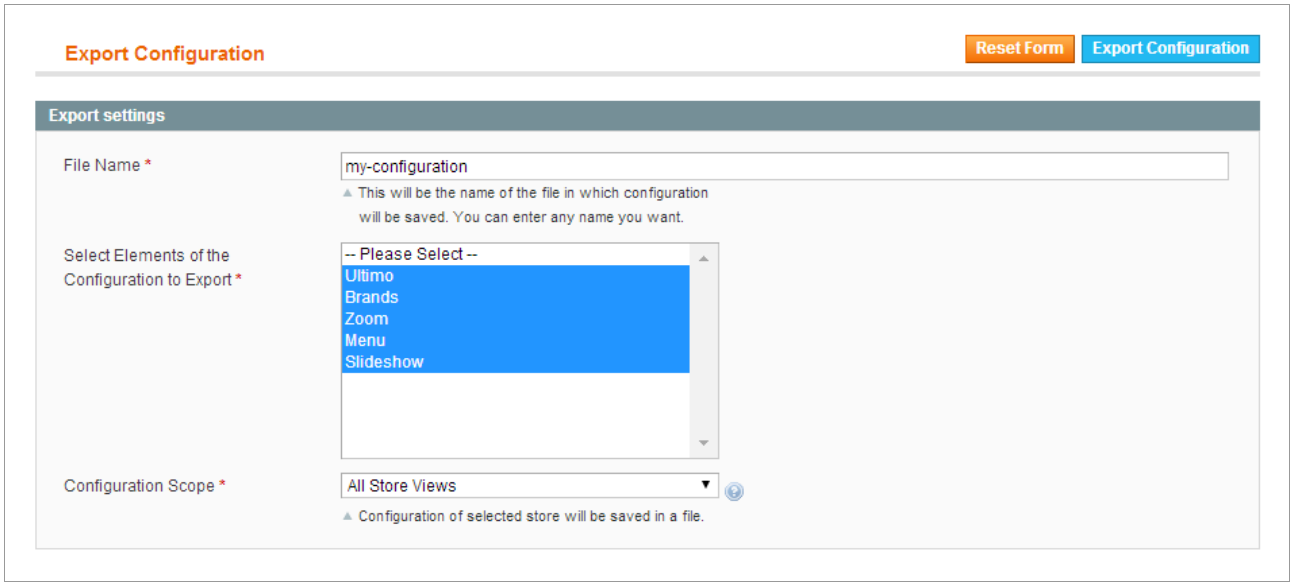
5. If you want to change the imported configuration settings, all the settings can be found in section *System > Configuration*. To make changes in the configuration, **you need to select that scope** using the **Current Configuration Scope** switcher which can be found in the top left corner of *System > Configuration* section – see the screenshot below:



For example, if you imported to the “French” store view, you need to select that scope using the **Current Configuration Scope** switcher. **Otherwise your changes will not take any effect.**

- ## Export

  – click this button to display data export form. You can export theme settings from selected sections. Specify the name of the file in which configuration will be saved. This name will appear on the configuration import form which was described earlier in this chapter so you will be able to import your configuration any time you want.



Exported configuration will be saved in a file in the following directory:
`app/code/local/Infortis/Ultimo/etc/importexport/config/`

Import Static Blocks

- **Import Static Blocks** – click this button to import all predefined static blocks provided with this theme. Refer to **6.2 Built-in CMS blocks** for more details about the static blocks. You can find there the list of identifiers of all blocks.

  To reset (re-import) selected static blocks, delete blocks which you want to reset, and then import blocks again. This way you will only import blocks which are missing.

> **Important:**
>
> in case you need the original content of some static blocks, XML file with all static blocks can be found in the following directory:
> `app/code/local/Infortis/Ultimo/etc/import/`

- **Overwrite Existing Blocks** – if set to **Yes**, imported blocks will overwrite existing blocks with the same identifiers (if any blocks with the same identifiers already exist). You can use this option when you want to restore the original content of all blocks.

> **Note:**
>
> to change this option select **Yes** or **No** and then **save the configuration**. Without saving, the option will remain unchanged after you refresh the page.

Import Pages

- **Import Pages** – click this button to import sample CMS pages provided with this theme.

  Pages can be edited in `CMS > Pages`. Names of all provided pages are prefixed with the name of the theme. Home page configuration is described in this chapter: **6.1 Home page**

  To reset (re-import) selected pages, delete pages which you want to reset, and then import pages again. This way you will only import pages which are missing.

> **Important:**
>
> in case you need the original content of CMS pages, XML file with all pages can be found in the following directory:
> `app/code/local/Infortis/Ultimo/etc/import/`

- **Overwrite Existing Pages** – If set to **Yes**, created pages will overwrite existing pages with the same identifiers (if you have any). You can use this option when you want to restore the original content of all pages.

> **Note:**
>
> to change this option select **Yes** or **No** and then **save the configuration**. Without saving, the option will remain unchanged after you refresh the page.

8. Theme Design

This chapter refers to the following section of the theme admin panel:
`System > Configuration > Ultimo > Theme Design`

Ultimo is equipped with administrative module which gives you the ability to configure the visual appearance of your store. You can change the colors of many elements by using color pickers or by specifying the color code manually. You can also apply textures (patterns) for the main sections of the page.

Names of the most of the options are pretty self-explanatory. Some of them are described in this chapter.

> **Note:**
>
> **disable Magento cache** before you start configuring your store. Otherwise you will need to flush the cache in order to see any changes you have made in the config.

> **Important:**
>
> after creating new store view (or after changing the store view code), re-save theme settings (clicking **Save Config** button) in `System > Configuration > Ultimo > Theme Design` and in `System > Configuration > Ultimo > Theme Layout`. Otherwise design/layout settings will not be applied to that new store view.

8.1 Colors

In this tab, you can configure basic colors for the entire page. These colors will be inherited by all sections of the page.

Basic Colors

Text and links.

Buttons

Buttons such as “Add to cart”, “Log In” etc.

Interface Icons

Small buttons with icons, e.g.: arrows in product sliders, “sort order” in category view etc.

Custom Icons

Customizable icons (e.g. social icons) which can be used to build custom content.

Horizontal Links

Inline links, e.g. Magento's default Top Links in the header (set of links: Account, Wishlist, Log In), Footer Links etc.

Product Labels

Product labels “New” and “Sale”.

Other

Other elements not classified elsewhere.

8.2 Font

In this tab, you can configure the options related to fonts.

8.3 Effects

In this tab, you can set additional effects for some of the page elements.

- Content Area Shadow – If set to **Yes**, all sections with nontransparent **Inner Background Color**, will drop soft shadow.

  For example, if you set any color in **Inner Background Color** filed in **Footer – Primary** tab, shadow will be displayed in the main area of the footer section.

- Main Menu Shadow – select which container of the menu will drop shadow. This field will be visible if Content Area Shadow was enabled.

> **Note:**
>
> unlike other sections, menu doesn't need to have **Inner Background Color** to drop shadow.

- Structural Elements Transition – if enabled, main structural elements of the page layout will be animated during web browser window resizing. It is recommended to leave this option disabled for browser compatibility.

8.4 Page

In this tab, you can configure the options related to the background of the entire page.

- Background Color – background color of the entire page.

- Background Image – here you can upload an image which will be displayed as a background of the entire page. Click the **Browse...** button and locate the image on your computer.

> **Note:**
>
> sample backgrounds can be found in `media/wysiwyg/infortis/ultimo/custom/background` directory.

Once configuration has been saved, a small icon of the existing image will appear to the left of the button. You can then configure the way the image is displayed with the following options:

- **Background Repeat** – defines how the background image is repeated. It can be repeated along the vertical axis, the horizontal axis, both, or not repeated at all. By default it is repeated both vertically and horizontally.
- **Background Attachment** – determines whether the background image is fixed or scrolls with the rest of the page. By default it scrolls.
- **Background Position (x-axis)** – defines the initial position of the background image along the horizontal axis. By default it is aligned to the center of the page.
- **Background Position (y-axis)** – defines the initial position of the background image along the vertical axis. By default it is aligned to the top of the page.

- **Pattern** – background pattern of the entire page. Pattern covers the background image.

8.5 Header

In this tab, you can configure the options related to the header section of the page. Background image can be set in exactly the same way as for entire page (which is described in previous chapter).

8.6 Header – Top

In this tab, you can configure the area at the top of the header. Settings from this tab will override settings from the `Header` tab.

8.7 Header – Primary

In this tab, you can configure elements of the primary area of the header.

8.8 Main Menu

In this tab, you can configure the options related to the main menu bar.

8.9 Main

In this tab, you can configure the options related to the main section of the page (the section located between the header and the footer).

8.10 Category View

In this tab, you can configure selected elements of the category view.

8.11 Product Page

In this tab, you can configure the product page (in Magento also called "product view").

8.12 Slideshow

In this tab, you can configure elements related to slideshows.

8.13 Footer

In this tab, you can configure the entire footer section of the page. Settings from this tab are inherited by all the other sub-sections of the footer (`Footer-Top`, `Footer-Primary` and `Footer-Bottom`). Selected settings can be overridden in sub-sections.

8.14 Footer – Top

In this tab, you can configure the options related to the area at the top of the footer. Settings from this tab will override settings from the `Footer` tab.

8.15 Footer – Primary

In this tab, you can configure the options related to the primary area of the footer. Settings from this tab will override settings from the `Footer` tab.

8.16 Footer – Secondary

In this tab, you can configure the options related to the secondary area of the footer (just below the primary area). Settings from this tab will override settings from the `Footer` tab.

8.17 Footer – Bottom

In this tab, you can configure the options related to the bottom area of the footer. Settings from this tab will override settings from the `Footer` tab.

9. Theme Layout

This chapter refers to the following section of the theme admin panel:
`System > Configuration > Ultimo > Theme Layout`

> **Note:**
>
> **disable Magento cache** before you start to configure your store. Otherwise you will need to flush the cache in order to see any changes you have made in the config.

> **Important:**
>
> after creating new store view (or after changing the store view code), re-save theme settings (clicking **Save Config** button) in `System > Configuration > Ultimo > Theme Design` and in `System > Configuration > Ultimo > Theme Layout`. Otherwise design/layout settings will not be applied to that new store view.

9.1 Responsive Layout

In this tab, you can configure the options related to the responsive layout.

- **Responsive Layout** – Enable or disable responsive layout.
- **Maximum Page Width** – maximum width of the content area of the page.

  To specify custom width select `Custom width...` from the drop-down list. New field will appear.
- **Custom Maximum Page Width** – you can specify custom maximum width of the page. To display this field you need to select `Custom width...` from the **Maximum Page Width** drop-down list. Enter value in pixels, do not add “px” suffix.
- **Fluid Width** – if enabled, the theme always uses maximum available space in the web browser window to display the content. If disabled, the width of the page is static and changes only on layout breakpoints when the web browser window is resized.

9.2 Disable responsive layout

In order to disable responsive layout, go to *System > Configuration > Ultimo > Theme Layout* section and set **Responsive Layout** field to `Disable`.

Some additional steps are also required:

- disable mobile version of the main menu: **Main Menu Mode** in *System > Configuration > Menu*.
- do not use `is_responsive` parameter in sliders: **6.1.4 Featured Products Slider**, **6.1.5 New Products Slider**, **12.2.4 Brand Slider**. With this parameter sliders are responsive and change the number of visible items depending on the width of the web browser's viewport
- remove collapsible elements from the footer static blocks (e.g. *block_footer_column1*), they are not needed if the layout is not responsive. List of footer static blocks can be found in **6.2 Built-in CMS blocks**.

CSS classes responsible for collapsible elements are described in **6.5.2 Collapsible blocks**. Basically you need to remove classes `collapsible` and `mobile-collapsible` from the content of the static blocks or add `active` class to the element's wrapper so that the element is not collapsed and hidden by default anymore.

10. Zoom

This chapter refers to the following section of the theme admin panel:
`System > Configuration > Zoom`

It describes the Cloud Zoom (which is an implementation of a free jQuery plugin Cloud Zoom http://www.professorcloud.com/) and Lightbox on product page (which is an implementation of a free jQuery plugin Colorbox http://www.jacklmoore.com/colorbox/).

10.1 Zoom configuration

Below you can find a list of the Zoom settings available in the admin panel.

10.1.1 General

- **Enable Module** – this option should be enabled in order to use zoom features on product pages. If disabled, images will be displayed in the same way as in Magento's "default" theme.
- **Use Cloud Zoom** – If set to **Yes**, Cloud Zoom will be used to zoom product images on product pages.
- **Position** – specify the position of the zoom area relative to the main image. If `Inside` is selected, zoom area will be placed inside the main image – useful if you would rather not obscure any other content.
- **Lens Opacity (%)** – specify opacity of the lens pointer.
- **Zoom Area Width** – width of the zoom area. Leave empty to use default (the width will be the same as the main image).

> **Note:**
> Field **Zoom Area Width** and **Zoom Area Height** will be ignored if **Position** is set to "Inside".

- **Zoom Area Height** – height of the zoom area. Leave empty to use default (the height will be the same as the main image).
- **Image Width After Zooming** – width of the enlarged image. This field refers to the actual resolution of the image after zooming.
- **Image Height After Zooming** – height of the enlarged image. This field refers to the actual resolution of the image after zooming.
- **Tint Color** – color which will cover the main image. Specify color in hexadecimal format, e.g. `#12bbcc` (not more than 7 characters). Leave empty to disable this effect. Does not work if **Soft Focus** is enabled.
- **Tint Opacity (%)** – specify opacity of the tint, where `0` is fully transparent, and `1` is fully opaque.
- **Soft Focus** – applies a subtle blur effect to the small image. Does not work if **Tint Color** effect is enabled.
- **Smooth Move** – Amount of smoothness/drift of the zoom image as it moves. The higher the number, the smoother/more drifty the movement will be. `1` = no smoothing.

10.1.2 Image Size

In this tab, you can override the default size of the product images.

- **Keep Image Aspect Ratio** – set **Yes**, to keep the aspect ratio of the product images on product page.

> **Note:**
> If set to **Yes**, height specified in **Main Image Height** and **Image Height After Zooming** will be ignored and the actual height of the images will be calculated automatically based on width.

- **Main Image Width** – specify width of the main image on product page.

> **Note:**
> Remember that uploading images smaller than the size specified in **Main Image Width** and **Main Image Height** fields may impact the quality of the images displayed on product page.

- **Main Image Height** – specify the height of the main image on product page. Leave empty to keep the aspect ratio of the image intact (**height will be calculated automatically based on width**). If the width is not specified, default width and height will be used.

10.1.3 Thumbnails

In this tab, you can enable a slider for product image thumbnails.

- **Use Slider** – enable/disable slider functionality for product image thumbnails.
- **Timeout** – to enable automatic scrolling of the slider, you need to specify the time (in milliseconds) between slide transitions. Leave it empty to disable automatic scrolling. Note that 1000 milliseconds = 1 second.
- **Speed** – duration (in milliseconds) of scrolling animation. Applied to scrolling with arrows (and touch swipe).
- **Automatic Scrolling Speed** – duration (in milliseconds) of scrolling animation. Applied to automatic scrolling (if **Timeout** was specified).
- **Loop** – if enabled, scrolling will loop.
- **Move Items** – number of thumbnails that should move on animation. Allowed values:
  - `1` – will move one thumbnail.
  - `0` – will move all visible thumbnails.
- **Lazy Loading** – delays loading of images. If enabled, images outside of viewport will not be loaded before user scrolls to them.

10.1.4 Lightbox

- **Enable Lightbox** – if enabled, lightbox functionality will be available for the main product image.
    - If the **Use Cloud Zoom** field is set to **Yes**, zoom button will be displayed above the main product image. After clicking the zoom button, the image will be displayed in the lightbox
    - If the **Use Cloud Zoom** field is set to **No**, the entire product image can be clicked to display the image in the lightbox
- **Maximum Width** – maximum width of the lightbox content. Leave empty to show image in its maximum size (specified in **Image Width After Zooming** field). You can enter values in the following formats:
    - 90%
    - 500px
    - 500
- **Maximum Height** – maximum height of the lightbox content. Leave empty to show image in its maximum size (specified in **Image Height After Zooming** field). Available formats: see the field above.
- **Lightbox As Gallery** – if set to **Yes**, after image is displayed in the lightbox, "next" and "previous" buttons will be displayed.

11. Slideshow

This chapter refers to the following section of the theme admin panel:
`System > Configuration > Slideshow`

Description of all the settings of the Slideshow available in the admin panel can be found later in this chapter: **11.3 Slideshow configuration**.

11.1 How to display a slideshow on any page

Slideshow can be displayed on any CMS page and in any static block. To display a slideshow in a selected place use the following block code:

```
{{block type="ultraslideshow/slideshow" template="infortis/ultraslideshow/slideshow.phtml"
slides="block_slide1, block_slide2"}}
```

Enter the code in the content field of your page, Magento will automatically replace the code with a slideshow during page rendering.

Parameter `slides` must contain a comma separated list of slides which will be displayed in your slideshow. Each slide is a simple **static block**, so with this parameter you just need to provide a list of static block identifiers (you can add as many slides as you need) and each of these static blocks will be displayed as a slide in the slideshow.

Slides can contain any text or HTML. For quick start there are some sample static blocks which can be imported after theme installation, refer to **11.2 Static blocks as slides** for more details.

There are also special settings which will let you quickly add a slideshow to the home page. You can read more about it later in this chapter: **11.1.2 Full width slideshow on home page**.

11.1.1 List of all parameters

```
{{block type="ultraslideshow/slideshow" template="infortis/ultraslideshow/slideshow.phtml" slides="X" banner="X"}}
```

Replace **X** with proper values of the following parameters:

- **slides** – comma separated list of static block identifiers, for example: `block_slide1, block_slide2, block_slide3`. These static blocks will be displayed as slides. **This field is required**.

- **banner** – an identifier of a single static block which contains additional banners (or any other content) displayed at the side of the slideshow. This field is optional.

11.1.2 Full width slideshow on home page

To display a slideshow on the home page, you don't need to use the block code. Just go to *System > Configuration > Slideshow* and configure options which can be found in sections named *Main Slideshow on Home Page*.

> **Note:**
>
> these sections are related only to the main slideshow on the home page which is displayed just below the main menu. Settings from these sections will not be applied to any other slideshows.

List of slides

In the *Slides* field (under *System > Configuration > Slideshow > General > **Main Slideshow on Home Page***) enter identifiers (separated by comma) of all static blocks which should be displayed as slides in the slideshow.

Then, save the configuration and flush Magento cache. Since now your blocks will be displayed in the slideshow on the home page. You can display as many slides as you want – simply create more static blocks and add their identifiers in the *Slides* field.

Additional banners

In the *Banners* field (under *System > Configuration > Slideshow > Banners > **Main Slideshow on Home Page***) you can also enter identifier of the static block which contains additional small banners (or any other custom content). These banners will be displayed at the side of the slideshow.

Note that you need to enter **identifier of a single block**, not a list of blocks.

Full width slideshow

In the same section you can also select position of the home page slideshow. Slideshow can be configured to take full available width of the page. List and description of all the settings available in the admin panel can be found in **11.3 Slideshow configuration**.

11.2 Static blocks as slides

Each slide in a slideshow is a simple **static block** which can contain any text or HTML (including images, links etc.). To create a static block, go to `CMS > Static Blocks`. Refer to **4.8.2 Static blocks** for more information about static blocks in Magento.

Slides

There are many predefined static blocks provided with this theme (refer to **Error: Reference source not found Error: Reference source not found** for more details about the static blocks import). The following predefined blocks are dedicated for slideshow and can be used as a starting point for your own slides: `block_slide1`, `block_slide2`, `block_slide3`. But of course you can create and use any other blocks.

> **Important:**
>
> when you're creating a static block, make sure to set it as **Enabled** and assign it to a proper store view. Otherwise the static block will not be displayed in the slideshow even if it's on the list of slides for that slideshow.

Additional banners

There is also a predefined block with additional side banners for the slideshow: `block_slideshow_banners`. Of course, you can create and use any other static block with custom content.

11.2.1 Size of slides and banners

For the default value of the **Maximum Page Width** (which is 1024 pixels) the minimum width of images (slides) in the slideshow is 941 pixels. More information about the **Maximum Page Width** field can be found in chapter **9. Theme Layout**. You can of course use larger images, they will be scaled down automatically to fit the available space.

> **Important:**

> If the height of images in all your slides is not the same, you may want to enable option **Smooth Height**, it will allow height of the slideshow to animate smoothly on each transition, which means that the height of the slideshow will adapt automatically to the height of your images.

If the **Maximum Page Width** was changed, you will need to adjust your images to your current width of the page. The following table shows the minimum width of slides for each available value of the **Maximum Page Width**:

Maximum Page Width	Minimum slide image width
1024 px	941 px
1280 px	1176 px
1360 px	1274 px
1440 px	1352 px
1680 px	1490 px

Slideshow with small banners at the side

The following table shows the size of slides and banners for each maximum page width:

Maximum Page Width	Size of slides (px)	Size of banners (px)
1024 px	696 x 310	226 x 103
1280 px	870 x 387	282 x 129
1360 px	943 x 420	306 x 140
1440 px	1001 x 446	325 x 148
1680 px	1102 x 491	358 x 163

11.2.2 Clickable images

To make the slide image clickable, you can use this HTML as a block content:

```
<a href="{{store url='about-us.html'}}">
        <img src="{{media url='wysiwyg/infortis/ultimo/slideshow/01.jpg'}}" alt="Sample slide" />
</a>
```

- Replace `about-us.html` with a path of any page in your store. Path is relative to the store URL. For example: if you need a link to `www.example.com/cell-phones`, simply replace `about-us.html` with `cell-phones`.
- Replace `wysiwyg/infortis/ultimo/slideshow/01.jpg` with a path to your own images located in `media` directory of your Magento.

> **Note:**
>
> Always turn off WYSIWYG editor if you want to paste copied HTML or CMS markup tags. Otherwise WYSIWYG editor can break the markup.

11.3 Slideshow configuration

Below you can find a list of all slideshow settings available in the admin panel in `System > Configuration > Ultimo > Slideshow`.

11.3.1 General

- **Transition Effect** – the effect to be used for transition between two slides.

> **Note:**
>
> Leave empty to use simple horizontal slide animation.

- **Timeout** – time (in milliseconds) between slide transitions. Enter `0` to disable automatic transition. Note that 1000 milliseconds is equal to 1 second.
- **Transition Speed** – duration (in milliseconds) of slide transition animation. Defines how fast the current slide is replaced with the next slide.
- **Smooth Height** – enable, if slides do not have the same height. It will allow height of the slideshow to animate smoothly.
- **Pause on Hover** – if enabled, when the mouse is over the slideshow the automatic transition functionality will pause.
- **Loop** – if enabled, animation will loop.
- **Hide Slides On Mobile Devices** – if set to **Yes**, slideshow will be hidden on mobile devices (if browser viewport width is less than 768 pixels). Important: banners can be hidden separately, see options below.

Main Slideshow on Home Page

In this section you can quickly configure and enable the main slideshow on home page. Settings from this section will be applied **only to the main slideshow on home page**. To display slideshows in other places use slideshow block code described earlier in this chapter.

- **Slides** – comma separated list of static block identifiers, for example: `block_slide1`, `block_slide2`, `block_slide3`. These static blocks will be displayed as slides. Any content can be put inside slides.

> **Important:**
>
> Leave this field empty to remove the main slideshow from the home page.

- **Position: Above Content Area, Full Width** – if enabled, the main slideshow will be displayed at the top of the main content area of the home page (just below the main menu). **Slideshow will take full available width**.

> **Note:**
>
> Do not enable slideshow in both positions at the same time.

- **Position: Above Content Area, Boxed** – if enabled, the main slideshow will be displayed at the top of the main content area of the home page (just below the main menu).

> **Note:**
>
> Do not enable slideshow in both positions at the same time.

- **Top Margin** – additional top margin for the slideshow. You can set margin to add extra space above the slideshow. Enter negative value to pull the slideshow up.

11.3.2 Banners

In this tab you can configure additional small banners displayed at the side of the slideshow.

- **Position of Banners** – select position of the banners.
- **Hide Banners On Mobile Devices** – if set to **Yes**, banners will be hidden on mobile devices (if browser viewport width is less than 768 pixels).

Main Slideshow on Home Page

In this section you can quickly add small banners to the main slideshow on home page. Banners configured in this section will be **applied only to the main slideshow on home page**. To add banners to other slideshow use proper parameter in the slideshow block code described earlier in this chapter.

- **Banners** – identifier of the static block which contains additional banners (or any other content) displayed at the side of the slideshow.

> **Important:**
>
> Leave this field empty to remove additional banners from the main slideshow on home page.

11.3.3 Navigation

In this tab you can configure slideshow navigation elements.

- **Pagination** – enable/disable and select style of pagination.
- **Pagination Position** – select position of the pagination.

12. Brands

This chapter refers to the following section of the theme admin panel:
`System > Configuration > Brands`

List of all settings available in the admin panel can be found later in this chapter:
12.2 Brands configuration.

12.1 Basic information

Below you can find some basic information about brand configuration.

> **Important:**
>
> after you finish configuration of all settings, remember to reindex the data (**4.2 Index management**) and flush the cache (**4.1 Caching system**).

12.1.1 Brand attribute

First of all, to show brands, you will need an attribute which will store the information about brands. By default there is an attribute `manufacturer` in Magento which you can use for this (but you can of course create your own attribute, if needed).

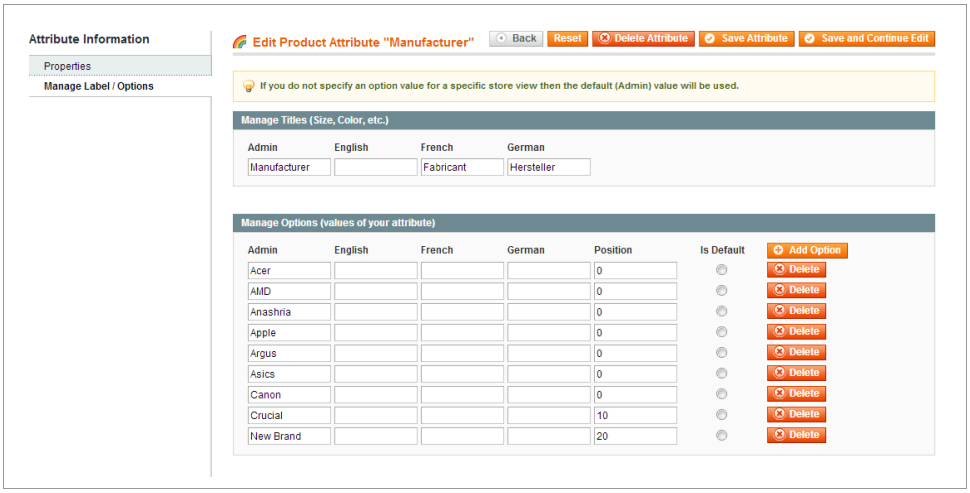
If you decide to create your own brand attribute, the most important thing is to select proper type. There are many types of attributes in Magento, but for brands you need to select `drop-down` in the **Catalog Input Type for Store Owner** field. This type of attribute will let you define a list of values (list of brands) available for products. For more information about attribute creation and management refer to **4.7 Attributes**.

12.1.2 Brand images

When your attribute is ready and was added to products, using your FTP client upload your brand logo images to the following directory: `media/wysiwyg/infortis/brands/` Optimal size of the image is 210 x 50 pixels. All images should have the same file extension (the extension can be specified in brand settings described later in this chapter).

12.1.3 Brand names and image file names

All images should be named as per the entries in the brand attribute, but the names of the files has to be simplified a little bit.



Below you can find the list of rules which you need to follow in order to convert a brand name to a correct file name. The module will follow the same rules to generate the path of the image file for each brand. If files are not named correctly, then the module will not be able to find and display brand images in your store.

The same rules will be used to generate links to custom pages if you configure brand logos to be links to custom pages.

> **Note:**
>
> in exactly the same way Magento simplifies category names and product names to automatically create **URL Key** for category pages and product pages. Full symbol conversion table can be found in Magento file: `app/code/core/Mage/Catalog/Helper/Product/Url.php`

Brand name conversion rules:

1. Replace all uppercase letters with lowercase letters, e.g.:

   for brand `BlackBerry` create an image named like this → `blackberry.png`

2. Define the word separator (for simplicity we will call it **the separator**) in the **Image URL Key Separator** field in *System > Configuration > Brands > General* section of admin panel. The separator will be used to replace all whitespace characters and other non-letter and non-digit characters in file names.

   It is recommended to use the default separator: a single hyphen (`-`). More info about separator can be found later in this chapter: **12.2 Brands configuration**.

3. Replace single space with the separator, e.g.:

   `Louis Vuitton` → `louis-vuitton.png`

4. Replace single special characters (non-letter and non-digit characters such as apostrophe, dot, ampersand, percent sign, exclamation mark etc.) with the separator, e.g.:

   `McDonald's` → `mcdonald-s.png`
   `AT&T` → `at-t.png`
   `Amazon.Com` → `amazon-com.png`
   `Coca-Cola` → `coca-cola.png`

5. Replace series of whitespace and special characters with a single separator, e.g.:

   `Marks & Spencer` → `marks-spencer.png`
   `J.P. Morgan` → `j-p-morgan.png`

6. Replace language specific symbols and characters (e.g. letters with diacritical marks) with letters from latin alphabet, e.g.:

   `Huáwéi` → `huawei.png`
   `Citroën` → `citroen.png`
   `Crédit Agricole` → `credit-agricole.png`

12.2 Brands configuration

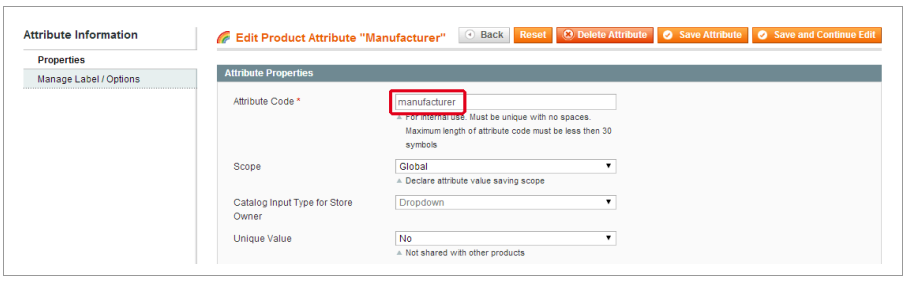
Below you can find a list of settings available in the admin panel *System > Configuration > Brands*.

12.2.1 General

- **Brand Attribute Code** – specify the **Attribute Code** of the attribute which stores the information about brands in your Magento (you can use Magento's default attribute `manufacturer` or create your own attribute).

  If such attribute doesn't exist in your store, you will need to create it and add it to the Attribute Sets which you use for your products.

  To find the **Attribute Code** of existing attribute go to *Catalog > Attributes > Manage Attributes*, click on the attribute and search the **Attribute Code** field:



- **Brand Image File Extension** – specify the file extension (jpg, png or gif) of your brand logo images.

- **Show Brand Image** – if set to `No`, brand image will be replaced with brand name (simple text). This option is useful when you don't want to show brand images, only the brand names.

- **Show Text If No Brand Image** –

  - if set to `Yes`: if brand image doesn't exist, it will be replaced with brand name (simple text).

- if set to `No`: if brand image doesn't exist, it will be not displayed and there will be no other information about brand.

Link

- Brand Logo is a Link to... – with this option, you can decide whether brand logo is a link (e.g. to search results page).
  - **No Link** – select this option if you don't want brand logo to be a link.
  - **Quick Search Results** – brand logo is a link to quick search results page. For example: logo of brand *Apple* on product page and in brand sliders will be a link to the following page:

    `www.example.com/catalogsearch/result/?q=apple`

    which displays a list of products returned by Magento quick search for term “Apple”.
  - **Advanced Search Results** – brand logo is a link to advanced search results page which displays a list of products returned by Magento advanced search for brand attribute value “Apple”.
  - **Custom Page** – brand logo can also be a link to custom pages (e.g. category view pages, CMS pages, pages created by 3rd-party extensions etc.). In that case, each brand logo on product page and in brand sliders will be a link to a custom page.

    > **Note:**
    >
    > this option doesn't create any custom pages for brands. It only makes it possible to link your brand logos to pages such as: category view pages, CMS pages or other pages created by 3rd-party extensions.

    After selecting this option, more settings will appear below this field.

- Base Path of Brand Pages – path specified in this field will be used to build a link to a custom brand page (for each brand on product page and in brand sliders). For example, brand logo can be a link to a category which contains products from that brand. It can also be a link to any CMS page you want. To use this option select `Custom Page` in **Brand Logo is a Link to...** field.

Path will be appended to the store base URL before the brand name.
For example: `www.example.com/custom/base/path/apple`

`www.example.com/`	`custom/base/path`	`/apple`
Store base URL	*Base Path of Brand Pages*	*Brand name*

Here you can see some examples.

- **Example 1:**

  You sell smartphones and you want *Apple* logo to be a link to a category with other Apple products.

  - You created top-level categories for all your brands. If the category page URL is *www.example.com/apple*, then you should leave the **Base Path of Brand Pages** field empty.

  - You created sub-categories for all your brands. They are sub-categories of the main category named “All Brands” and the URLs of category pages are the following:
    - *www.example.com/all-brands/apple*,
    - *www.example.com/all-brands/samsung* etc.
    In that case you should enter `all-brands` in the **Base Path of Brand Pages** field. Thanks to that the module will be able to build correct links to category pages for all your brands.

- **Example 2:**

  You sell smartphones and you want *Apple* logo to be a link to a custom CMS page with some additional info about the brand. In Magento custom pages can be easily created in *CMS > Pages*. For each page you need to specify value of **URL Key** field which is a unique identifier of a page. E.g., custom page with URL Key `apple` will have the following URL: *www.example.com/apple*

  - You created custom pages for all your brands. If the URLs of your custom pages are the following:
    - *www.example.com/apple*,
    - *www.example.com/samsung* etc.,
    then you should leave the **Base Path of Brand Pages** field empty.

  - URL Key of a custom page can be more complex than just a single word. All your brand pages can have some common part, e.g., `my-brands/apple`, `my-brands/samsung` etc. You created custom pages for all your brands.

The URLs of your custom pages are the following:
- *www.example.com/my-brands/apple*,
- *www.example.com/my-brands/samsung* etc.
In that case you should enter `my-brands` in the **Base Path of Brand Pages** field. Thanks to that the module will be able to build correct links to custom pages for all your brands.

- **Append Category URL Suffix** – if **Category URL Suffix** exists (in *System > Configuration > Catalog*), it can be appended at the end of the brand link. For example: if `.html` was set as the **Category URL Suffix**, logo of brand *Apple* on product page and in brand sliders will be a link to the following page:

  `www.example.com/custom/base/path/apple.html`

SEO Settings

- **URL Key Separator** – specify character which will be used as word separator in URLs of brand pages. Hyphen (`-`) is the recommended character. Other separators, such as underscores (`_`) or plus-signs (`+`) should be avoided. More information: *https://www.youtube.com/watch?v=AQcSFsQyct8*

  Module will strip all whitespace characters and other non-letter and non-digit characters and replace them with the separator. For example:

  if brand name is `Marks & Spencer` and separator is hyphen (`-`), the brand name in URL will be the following: `marks-spencer`

  More information about how brand names are converted (to generate name of the brand image file or brand page URL) can be found in **12.1.3 Brand names and image file names**.

- **Image URL Key Separator** – specify character which will be used as word separator in the names of brand image files. For example:

  if brand name is `Louis Vuitton`, separator is hyphen (`-`) and file extension is `png`, the name of the brand image file will be the following: `louis-vuitton.png`

> **Important:**
>
> the **Image URL Key Separator** field was added for backward compatibility. In next versions this field will be removed and the character specified in the **URL Key Separator** field will also be used in the names of brand image files.
>
> To prepare your store for this, we recommend to use hyphen (`-`) as the

separator for images. In previous versions underscore (_) was the default separator for images. So what you need to do is to enter hyphen in the Image URL Key Separator field and edit names of all your brand images: replace underscores with hyphens.

For example: change file name `louis_vuitton.png` to `louis-vuitton.png`

12.2.2 Brand on Product Page

- **Show Brand on Product Page** – if enabled, will show a brand logo on the product page (if product has an attribute which specifies the brand).

12.2.3 Brand List

Select Brands

In this tab you can select list of brands which will be displayed in brand sliders. If you have multiple sliders (e.g., on different pages of the store), then each slider will display the same list of brands.

> **Note:**
>
> you can also override the list of selected brands by using `brands` parameter in the slider block code – parameter is described later in this chapter. With parameter it is possible to display different set of brands in each slider.

- **All Brands** – if set to `Yes`, all existing brands will be displayed. If set to `No`, you will be able to select brands in the field below.
- **Selected Brands** – select brands which you want to display.

Filter Brands

- **Brands Assigned To Products** – if set to `Yes`, display only those brands, which are currently assigned to products.
- **Brands Assigned To Products In Stock** – if set to `Yes`, display only those brands, which are currently assigned to products which are in stock.

12.2.4 Brand Slider

Brand slider can be displayed on any CMS page and in any static block. For more datails see the block code which is described later in this chapter.

Below you can find a list of global settings for all brand sliders, some of the settings can be overridden per slider by using parameters which are described later in this chapter.

- **Show All Brands** – if set to **Yes**, slider will display all existing brands. Otherwise slider will display only those brands, which are currently assigned to products.
- **Timeout** – Time (in milliseconds) between slide transitions. Enter `0` to disable automatic transition. Note that 1000 milliseconds is equal to 1 second.
- **Speed** – duration (in milliseconds) of scrolling animation. Applied to scrolling with arrows (and touch swipe).
- **Automatic Scrolling Speed** – duration (in milliseconds) of scrolling animation. Applied to automatic scrolling (if **Timeout** was specified) and to scrolling with pagination.
- **Pause on Hover** – if enabled, when the slider is mouseovered then the automatic scrolling will pause.
- **Loop** – if enabled, scrolling will loop.
- **Lazy Loading** – delays loading of images. If enabled, images outside of viewport will not be loaded before user scrolls to them.

Brand slider block code and parameters

To display brand slider, enter the following code in the content field of any page or static block, Magento will automatically replace the code with a slider during page rendering:

```
{{block type="brands/brands" template="infortis/brands/brand_slider.phtml"
breakpoints="[0, 1], [320, 2], [480, 2], [768, 3], [960, 4], [1280, 5]" block_name="Our
Brands"}}
```

Here is the list of available parameters. Some of them can override global settings:

Required parameters:

- `block_name` – the name of the block displayed as a heading.

Optional parameters:

- `brands` – with this parameter you can specify the list of brands (separated by comma) which will be displayed in the slider. You can have multiple sliders and each can display completely different set of brands – just provide different list of brands in each slider code using this parameter, for example:

```
brands="Apple, Samsung, Louis Vuitton, Coca-Cola"
```

  This list will override the list of brands which was selected directly in the admin panel in *System > Configuration > Brands > Brand List* section. But if this parameter is not specified, slider will display brands selected in the admin panel.

- `breakpoints` – with this parameter you can specify the number of items in a row visible with a particular browser viewport width. Use the following format (this is the default value, it will be used if this parameter is not specified in the block code):

```
breakpoints="[0, 1], [320, 2], [480, 2], [768, 3], [960, 4], [1280, 5]"
```

  Value of this parameter consists of pairs of numbers in brackets separated by comma. In each pair of numbers `[A, B]` number `A` indicates the web browser viewport width, number `B` indicates the number of items visible if browser viewport width is greater that the width specified in `A`.
  In other words: display `B` items if viewport width is above `A` pixels. Below you can find some examples:

  - **Example 1:**

```
breakpoints="[0, 2], [480, 3], [960, 5]"
```

    which can be translated as: display 2 items if viewport width is between 0px and 480px, display 3 items if viewport width is above 480px, display 5 items if viewport width is above 960px.

  - **Example 2:**

```
breakpoints="[0, 4]"
```

which can be translated as: always display 4 items.

- **Example 3:**

```
breakpoints="[0, 3], [960, 6]"
```

which can be translated as: display 3 items on narrow screens (if viewport width is below 960px), and display 6 items on wider screens (above 960px).

- `is_responsive` – enter `0` if the slider **should not be responsive**. If this parameter is not specified, it is always assumed that the slider is responsive.

> **Important:**
>
> If the slider **is not** responsive, the number of visible items should be specified with the additional **show_items** parameter (see below).

  - *`show_items`* – the number of products in a row in the non-responsive slider. Use this parameter only if responsive behavior was disabled with parameter `is_responsive` (see above)

- `timeout` – to animate the slider automatically, specify the time (in milliseconds) between transitions. Note that 1000 milliseconds = 1 second.
- `move` – number of items that should move on animation. Allowed values:
  - enter `1` to move one item.
  - enter `0` to move all visible items. This is also the default value which is used if the parameter is not specified.
- `loop` – enter `1` if the slider should loop.
- `pagination` – enter `1` to display pagination.

13. Menu

This chapter refers to the following sections of the admin panel:

- General menu settings:
  `System > Configuration > Menu`

  Description of general settings can be found in this chapter:
  13.1 Menu configuration

- Menu settings which can be applied individually for each category:
  `Catalog > Category Management`, in the `Menu` tab.

  Description of individual category settings can be found in this chapter:
  13.2 Drop-down boxes

More information about category management in Magento can be found in chapter **4.5 Categories**.

> **Note:**
>
> in Magento you can limit number of category levels displayed in the menu. Refer to **4.5.3 Depth of the menu** for more details.

13.1 Menu configuration

Below you can find a list of the menu settings available in the admin panel `System > Configuration > Menu`.

13.1.1 General

- **Main Menu Bar** – enable/disable main menu bar in the header section of the page.
- **Main Menu Mode** – select menu mode.
  - **Drop-down** – standard drop-down menu.
  - **Mobile** – collapsible accordion dedicated for mobile devices.
  - **Drop-down/Mobile** – combination of both modes. You can specify the width of the browser viewport, below which the drop-down menu turns into the mobile menu (see the **Threshold** field).
- **Show Categories** – enable to display categories in the main menu.

13.1.2 Menu Bar

Home Link: Single Icon

- Single Icon as Home Link – enable to show single icon as a link to the home page. Icon will be displayed without any additional background color. It can be used instead of the standard menu item with a link to the home page (see below).

Home Link: Inside Menu Item

- Home Link: Text – if enabled, will display menu item with text "Home" as a link to the home page. This link will be displayed with the same colors as other menu items.

- Home Link: Icon – if enabled, will display menu item with icon as a link to the home page. This link will be displayed with the same colors as other menu items.

Additional Settings

- Align Drop-down With Menu Bar – if set to **Yes**, a drop-down box of the top-level category will be aligned with the right edge of the menu bar if the right edge of the drop-down box sticks out of the menu bar area (i.e. if it's further to the right than the menu bar). If a drop-down box is wider than the menu bar, width of the drop-down box will be automatically reduced to match with the width of the menu bar.

  If set to **No**, a drop-down box can stick out of the menu bar area.

- Custom Block - Align Right – if set to **Yes**, static block `block_header_nav_dropdown` in the main menu will be aligned with the right side of the menu bar.

> **Note:**
>
> this block is deprecated and will be removed in future versions. There is a better method to add custom drop-down boxes to the menu, for more details refer to **13.4 Custom drop-down boxes in the menu**.

13.1.3 Mobile Menu

- **Threshold** – width of the browser viewport (in pixels), below which the drop-down menu turns into a mobile menu (collapsible accordion) dedicated for mobile devices.

> **Important:**
>
> Threshold is needed only if the menu is in `Drop-down/Mobile` mode. Values below 960 are not recommended if you have very wide drop-down boxes with custom content in your menu – wide drop-downs will not look good on narrow screens such as small tablets.

- **Collapsed** – if set to **Yes**, mobile menu will be collapsed on page load.

  If set to **No**, mobile menu will not be collapsed and all the top-level items will be visible.

- **Show Category Blocks From X Levels of Categories** – specify number of levels of categories (counting from the top level) for which category blocks will be displayed in mobile menu.

  E.g. enter `1` to display blocks of 1st-level categories, enter `2` to display blocks of 1st-level and 2nd-level categories, enter `3` to display blocks of 1st-level, 2nd-level and 3rd-level categories, and so on.

  The allowed range is from `0` to `5`.

> **Important:**
>
> enter `0` to hide blocks of all categories on all levels in the mobile menu.

- **Hide Category Blocks Below 320px** – category blocks will be hidden in the mobile menu on small mobile devices (if width of the browser viewport is below 320px).

13.1.4 Sidebar Menu

In this section you can enable and configure the sidebar menu: vertical list of categories which can be displayed at the top of the sidebars on selected pages.

> **Note:**
>
> you can also display the sidebar menu inside any static block and on any CMS page by using the sidebar menu block code. The code is described later in this chapter.

Enable Sidebar Menu

Here you can enable/disable sidebar menu in different locations of the store. Names of the options are self-explanatory so we only show two of them as an example:

- Category View (Left Sidebar) – enable/disable menu at the top of the left sidebar in the category view above Magento's standard Layered Navigation. For more details about Layered Navigation refer to **4.5.4 Layered Navigation ("Shop By" filters)**.
- Category View (Right Sidebar) – enable/disable menu at the top of the right sidebar in the category view.

Basic Settings

- Block Title – the name of the block displayed at the top of the block.

  Additionally you can use variable `[current_category]` inside the title to create dynamic name of the block. Variable will be replaced with the name of the current category. On pages other than category view or product view this variable will be empty.

- Parent Category – the menu will only contain children (subcategories) of the selected parent category. Available options:

  - **Root** – show all top-level categories in the store.
  - **Parent of current category** – show the current category and its siblings (other categories from the same level as the current category).

  - **Parent of current category (no siblings)** – show the current category. Other categories from the same level as the current category will be ignored.

  - **Current category** – show subcategories of the current category.

- Maximum Depth – this parameter controls how many levels in the hierarchy of categories (starting from the parent category) are to be included in the menu. Menu with a Maximum Depth of `1` will just be an un-nested list of categories from a single level (with none of the subcategories displayed). The default value is `0`, it will display all subcategories.

- Fallback To Root Category – if current category doesn't exist, go back to root category. This option is useful if sidebar menu should be displayed on pages with no current category (on pages other than category view). If set to “Yes” and if current category doesn't exist, the menu will behave like **Parent Category** field was set to “Root” (it will display top-level categories). If set to "No", menu will not be displayed at all if there's no current category.

  For example, if menu is displayed on the home page (**Parent Category** field is set to “Current category”) and **Fallback To Root Category** field is set to “Yes”, then the menu will display top-level categories.

  If in the same example **Fallback To Root Category** field is set to “No”, then the menu will not show any categories and will not be displayed at all.

- Fallback Block Title – this block title will be used instead of the standard block title, if **Fallback To Root Category** field is set to “Yes” and if current category doesn't exist.

- Show Number of Products – show number of products next to each category.

Additional Settings

- Hide Sidebar Menu on Mobile Devices – list of categories in the sidebar can be hidden on mobile devices (if browser viewport is narrower than 768 pixels). All categories are displayed in the top main menu by default so with this option you can save some space on mobile devices.

- Hide Magento's Category Filter – if you enabled sidebar menu, you may want to disable category filter in Magento's Layered Navigation block ("Shop By") in category view.

Sidebar menu block code

You can also display the sidebar menu inside any custom static block (for example, in the right sidebar: `block_right_top`). To display the menu, type in the menu code inside the static block content (Magento will automatically replace the code with the menu in the frontend), save the static block and flush Magento cache. Here's an example of the sidebar menu block code:

> **Important:**
>
> This block code has changed in version 1.4.0. If you used the code in previous versions, you need to replace the old code with the new one.

```
{{block type="ultramegamenu/navigation" parent="X" depth="X" block_name="X"
template="infortis/ultramegamenu/categories.phtml"}}
```

There are several parameters that can help you configure the menu to suit your needs. Replace **X** with proper values of the parameters:

- **parent** – the menu will only contain children (subcategories) of the selected parent category. This parameter is similar to field **Parent Category** which is described earlier in this chapter. Available options:
  - `root` – show all top-level categories in the store.
  - `parent` – show the current category and its siblings (other categories from the same level as the current category).
  - `parent_no_siblings` – show the current category. Other categories from the same level as the current category will be ignored.
  - `current` – show subcategories of the current category.
  - Or specify the ID (an integer number) of any existing category.
- **depth** – specify how many levels in the hierarchy of categories (starting from the parent category) are to be included in the menu. This parameter is similar to **Maximum Depth** in the admin panel.

- **block_name** – the name of the block displayed as a heading.

Examples

Here you can see some examples of the block code:

```
{{block type="ultramegamenu/navigation" parent="root" depth="2" block_name="All Categories in
our store" template="infortis/ultramegamenu/categories.phtml"}}
```

```
{{block type="ultramegamenu/navigation" parent="parent" depth="1" block_name="Current level of
categories" template="infortis/ultramegamenu/categories.phtml"}}
```

```
{{block type="ultramegamenu/navigation" parent="parent_no_siblings" depth="3"
block_name="Current category with subcategories"
template="infortis/ultramegamenu/categories.phtml"}}
```

```
{{block type="ultramegamenu/navigation" parent="current" depth="4" block_name="Subcategories of
current category" template="infortis/ultramegamenu/categories.phtml"}}
```

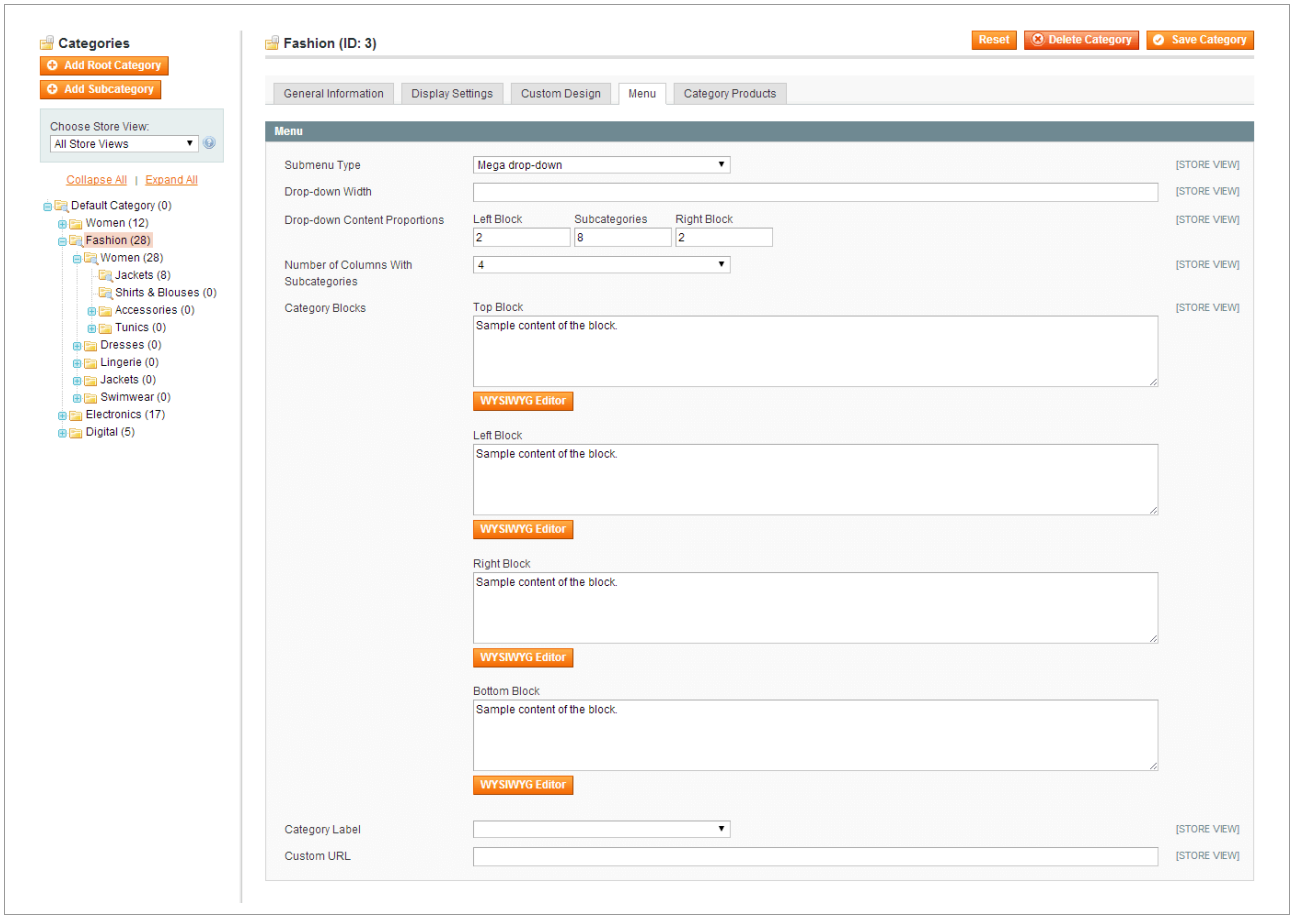
13.1.5 Category Labels

- **Label 1** – define the text of the category label 1. Labels can be assigned to every category in `Catalog > Manage Categories`.
- **Label 2** – define the text of the category label 2. Labels can be assigned to every category in `Catalog > Manage Categories`.

13.2 Drop-down boxes

13.2.1 Drop-down box configuration options

Each category in the menu can have a drop-down box containing subcategories or any other custom content. Below you can find a list of the drop-down box settings available in the admin panel. To configure the drop-down, go to `Catalog > Category Management`, select a category and open the `Menu` tab:



- **Submenu Type** – if category has subcategories, choose how subcategories should be displayed. You can choose the type individually for each category. For more details refer to **13.2.2 Submenu types**.

- **Drop-down Width** – in this field you can override the default width of the drop-down box. Width can be specified:
  - in pixels, for example: `150px`,
  - or as a percentage, for example: `200%` (percentage of the containing block's width).

  In top-level categories (and only in top-level categories) if **Submenu Type** is "Mega drop-down" width of the drop-down box can also be specified in grid units (number between 1 and 12), for example: enter `4` and the drop-down box will have a width of 4/12 of the entire menu bar.

  Default width of drop-downs:

  - The mega drop-down of a top-level category is by default as wide as the menu bar.
  - The mega drop-down of a category on lower levels is by default as wide as the menu item of that category. So it is necessary to override the default width. For example, if you enter `300%`, the drop-down will be three times as wide as the category item.
  - The classic drop-down is by default around 230 pixels wide.

- **Drop-down Content Proportions** – proportions between three sections of the mega drop-down box: **Left Block**, block of subcategories, **Right Block**.

  Proportions are specified in grid units. For each section enter a number between 1 and 12. Sum of the grid units entered for all three sections has to be equal 12.

> **Note:**
>
> category blocks are displayed only in mega drop-down boxes so this option is applicable only for categories with "Mega drop-down" selected in the **Submenu Type** field.

- **Number of Columns With Subcategories** – in this field you can select the number of columns in which the subcategories are displayed in the mega drop-down box. For example, select `3` to display subcategories in three columns. Default value is `4`.

> **Note:**
>
> subcategories are displayed in columns only in mega drop-down boxes so this option is applicable only for categories with “Mega drop-down” selected in the **Submenu Type** field (in the classic drop-downs subcategories are displayed one below the other).

- Category Blocks – if “Mega drop-down” was selected in the **Submenu Type** field, you can display additional content in the drop-down box.

  Content can be inserted into four blocks which are described below. To distinguish these blocks from other blocks in Magento, we will call them **category blocks**. For more details refer to **13.2.3 Category blocks**.

  If you don't want to display any additional content in the drop-down box, simply leave these four fields empty:

  - Top Block – category block displayed above the subcategories.
  - Left Block – category block displayed at the left side of the subcategories.
  - Right Block – category block displayed at the right side of the subcategories.
  - Bottom Block – category block displayed below the subcategories.

> **Note:**
>
> if you need to split your content inside category blocks into columns, you can use the built-in grid system – refer to **6.4 Grid system** for more details about the grid.

- Category Label – eye-catching labels for categories. Labels needs to be defined in `System > Configuration > Menu > Category Labels`. Refer to **13.1.5 Category Labels** for more details.
- Custom URL – enter a single hash character (`#`) to make the category **not clickable** in the main menu. It is especially useful when you want to create a drop-down box in the menu (to display some custom content) but don't want to redirect customers to any specific category.

  Leave this field empty if no changes are needed and the category in the menu has to be a link to the category page.

You can also use this field **to replace the default link of the category** with any custom link. To do this, enter custom URL path into this field. The path will be appended to the base URL of the store to create a new link. For example, enter `contacts/` to create a link to Magento's default “Contact Us” page, or `customer/account/create/` to create a link to “Create an Account” page.

More info about adding custom links to the menu can be found in **13.3 Custom links in the menu**.

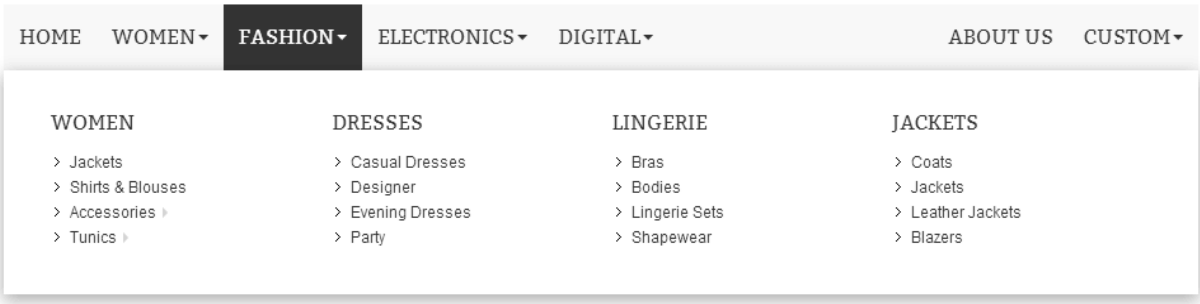
13.2.2 Submenu types

Available types

If the category has subcategories (i.e. it has a submenu), you can choose how the subcategories will be displayed. Go to *Catalog > Category Management*, select a category and open the *Menu* tab. There are three available options in **Submenu Type** field:

1. **Mega drop-down**

   Wide drop-down box which can display subcategories in multiple columns. It can also display additional content from the category blocks (which are described later in this chapter).



2. **Classic drop-down**

   Simple drop-down box which displays subcategories one below the other. It can display any number of levels of categories. Classic drop-down is the default type and it will be used if no type was specified for the category.

3. Simple submenu (no drop-down)

Additional type which can be used to display subcategories not in a drop-down box, but just below the category as a simple bulleted list.

> **Note:**
>
> this option can't be applied to top-level categories.

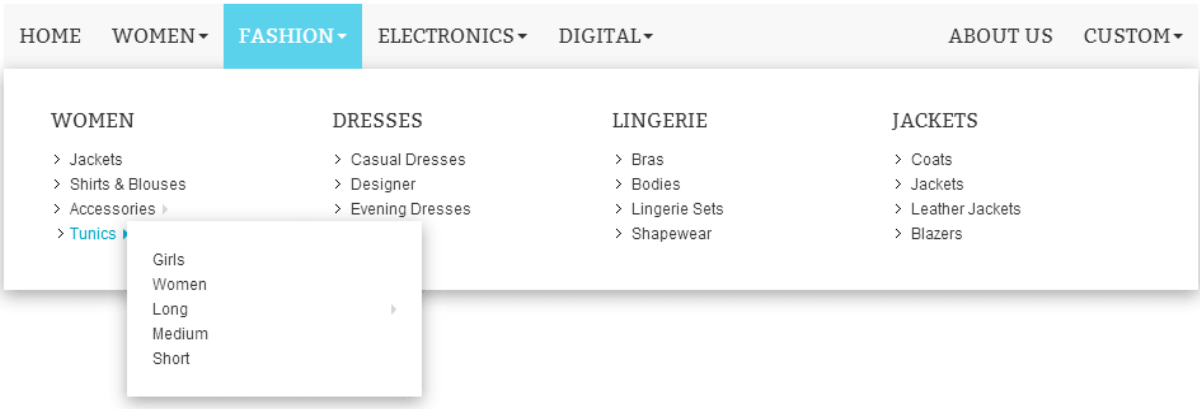
If in the previous example we will change the type of the “Dresses” category to “Simple submenu (no drop-down)”, the drop-down box will look like this:

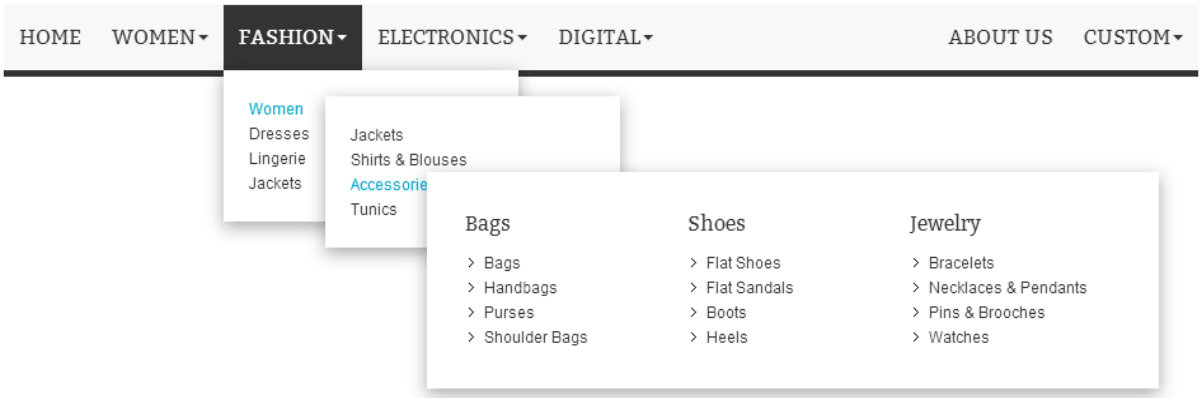
So the subcategories of the “Dresses” category will be displayed as a simple bulleted list.

Multilevel menus

Multilevel menus are fully supported, categories inside a drop-down box can have their own subcategories which will also be displayed in drop-down boxes.

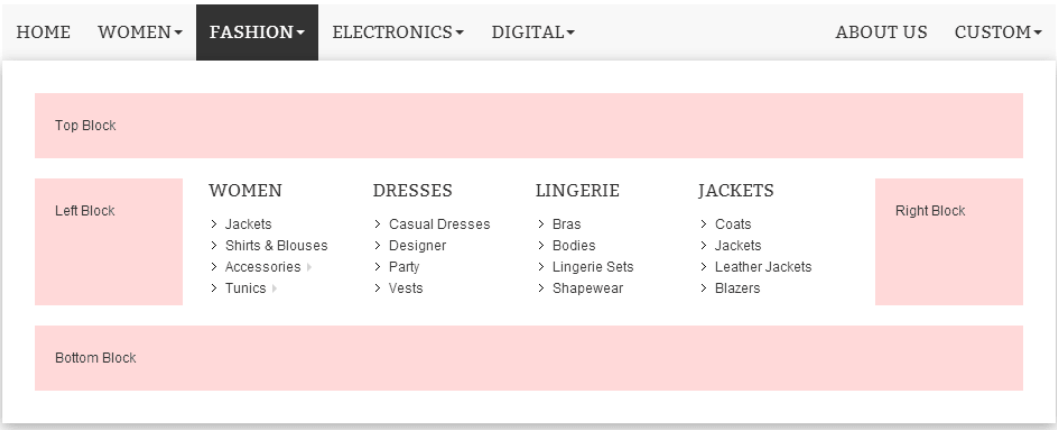
Additionally, all of the submenu types can be mixed together. For example, a classic drop-down can be nested inside a mega drop-down, and a mega drop-down can have classic drop-downs inside. See examples below:



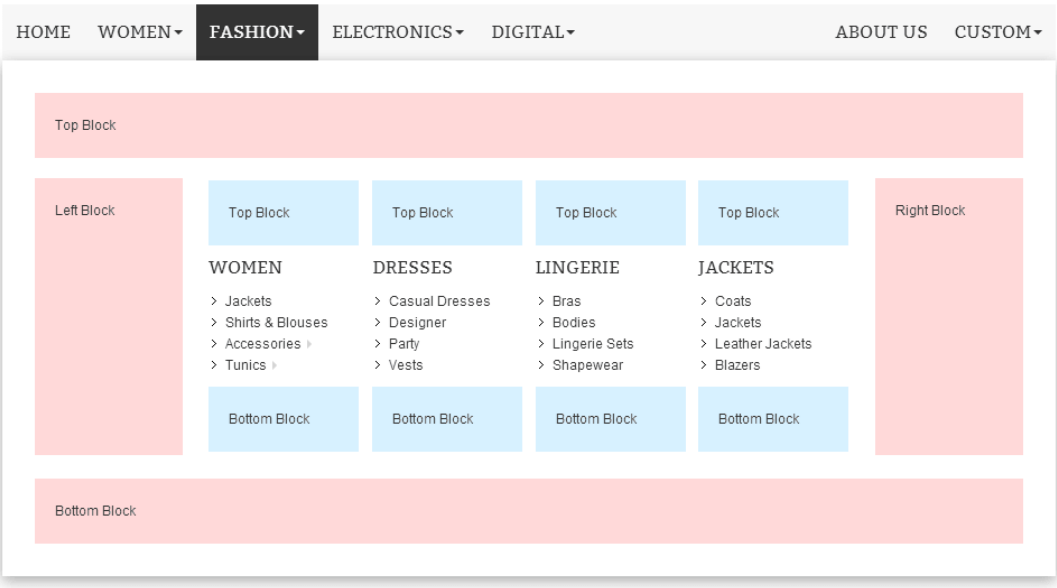


13.2.3 Category blocks

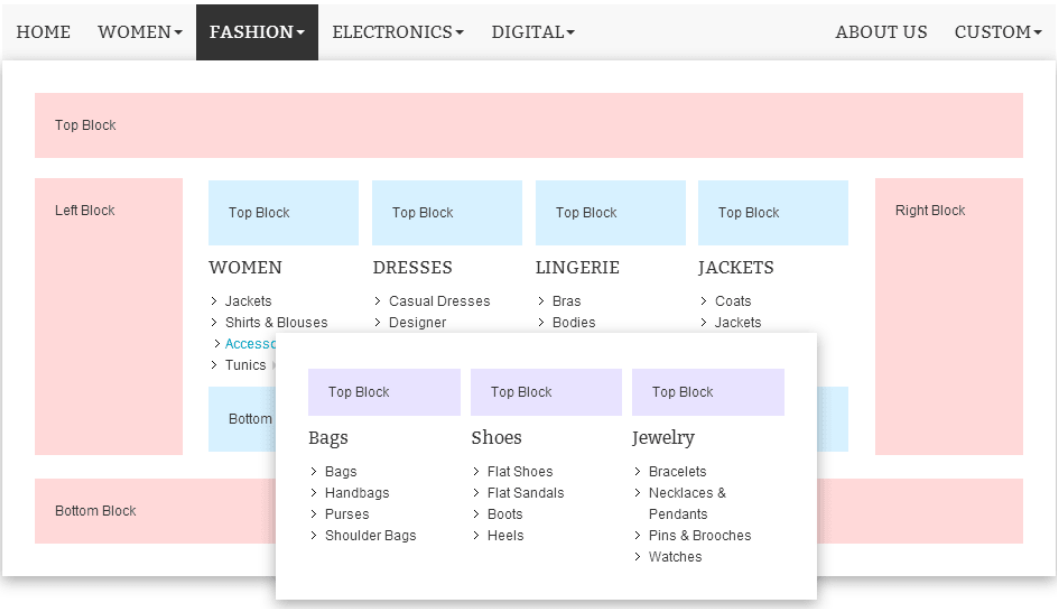
If “Mega drop-down” was selected as the submenu type, you can display additional content inside the drop-down box. Custom content can be inserted into four **category blocks**. On the example below, blocks of the “Fashion” category are marked with light red:



Mega drop-downs can also display category blocks of lower categories. For example, if the subcategories of the “Fashion” category (“Women”, “Dresses”, “Lingerie” and “Jackets”) have any content inside Top Block and Bottom Block, these blocks will be displayed in the drop-down. On the example below, these blocks are marked with light blue color:

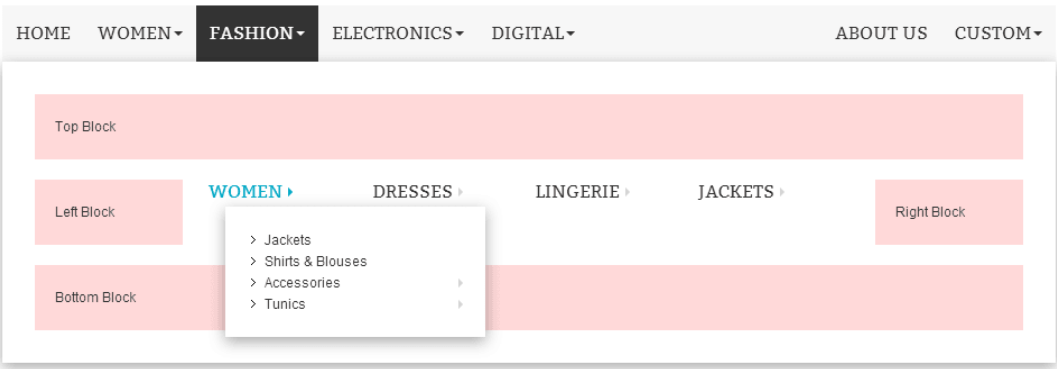


Moreover, subcategories of the “Fashion” category can have their own drop-down boxes. For example, if the “Accessories” category is configured as a mega drop-down (i.e. “Mega drop-down” was selected in the **Submenu Type** field), then a drop-down box will be displayed on mouse hover over the “Accessories” category. That drop-down can also contain its own category blocks with custom content (on the example below, these blocks are marked with light purple color):



Note that in this example, categories “Women”, “Dresses”, “Lingerie” and “Jackets” can't have “Mega drop-down” selected in the **Submenu Type** field (if it was selected, it will be ignored) because their subcategories are already displayed in the mega drop-down.

But categories “Women”, “Dresses”, “Lingerie” and “Jackets” **can have** “Classic drop-down” selected in the **Submenu Type** field. If they are configured this way, menu will look like this:

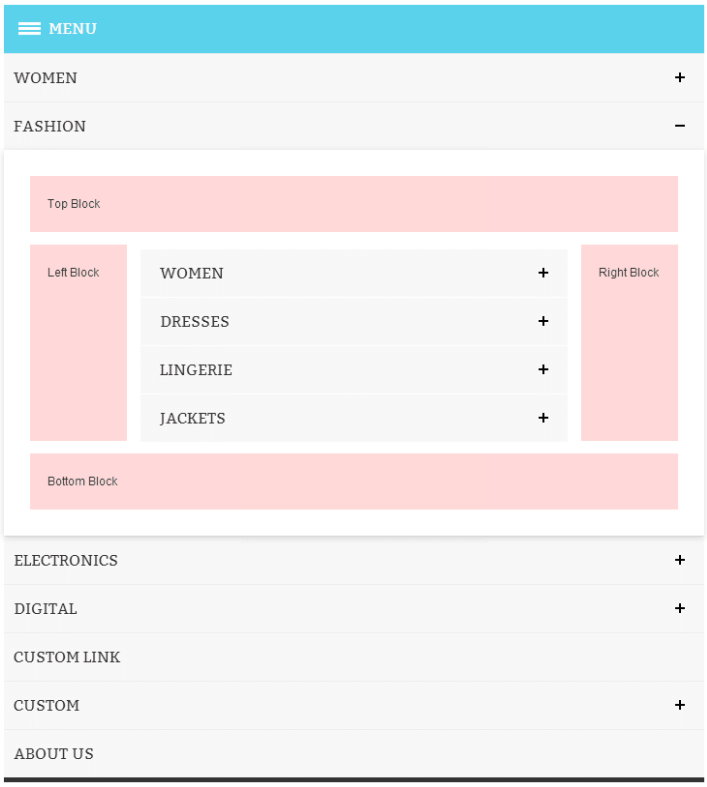


13.2.3.A Category blocks in mobile menu

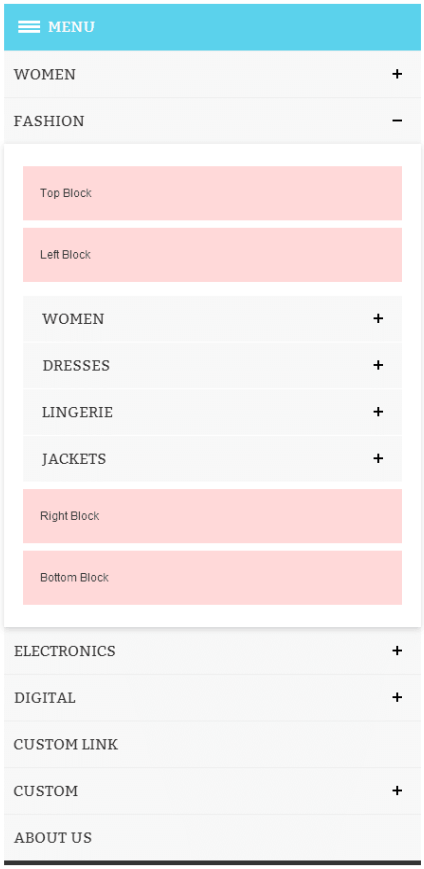
If the mobile menu is enabled, category blocks will also be displayed in the mobile menu (picture 1.). Mobile mode is activated when browser viewport is narrower than the threshold specified in the menu settings.

If screen is narrower than 768 pixels, category blocks will be displayed one below another (picture 2.), even the **Left Block** and the **Right Block**.

Picture 1.



Picture 2.



To hide category blocks in the mobile menu, you can use the **Show Category Blocks From X Levels of Categories** option which can be found in `System > Configuration > Menu > Mobile menu`. Refer to **13.1.3 Mobile Menu** for details.

You can also hide selected portions of the content inside the category blocks. There are two special classes:

- `hide-in-mobile-menu` – this class will hide the element when the menu is in the mobile mode
- `hide-in-desktop-menu` – this class will hide the element when the menu is in the standard mode.

Example below shows how to use these classes. Add this code inside one of the category blocks, e.g. **Top Block**:

```
<div class="hide-in-mobile-menu">
   This text will not be visible in mobile menu
</div>

<div class="hide-in-desktop-menu">
   This text will not be visible in desktop menu
</div>
```

13.2.4 Sample content for category blocks

Here you can find some samples of the content from category blocks:

Left Block

```
{{block type="ultimo/product_list_featured"
template="catalog/product/list_featured_slider.phtml" category_id="85" product_count="4"
breakpoints="[0, 1], [480, 2], [768, 1], [960, 2]" timeout="4000" move="1" centered="1"
size="size-s" hide_button="1" img_width="110" block_name="Recommended Bags"}}
```

Right Block

```
<a href="{{store direct_url='dress7.html'}}"><img class="fade-on-hover" src="{{media
url='wysiwyg/demo/infortis/ultimo/menu/01.jpg'}}" alt="Sample custom content of the category
block" /></a>
<br/><br/>
<h3>Featured Product</h3>
<p>This is a custom block ready to display any content. You can add blocks to any category in
the catalog...</p>
<a href="{{store direct_url='dress7.html'}}" class="go">View Product</a>
```

Top Block

```
<div class="show-separators">
     <ul class="links">
             <li class="label">Recommended:</li>
             <li class="first">
                     <a href="{{store
direct_url='electronics/smartphones/phone3.html'}}">Configurable Phone</a>
             </li>
             <li>
                     <a href="#">Galaxy S3</a>
             </li>
             <li>
                     <a href="#">Galaxy S4</a>
             </li>
             <li>
                     <a href="#">iPad Retina</a>
             </li>
             <li class="last">
                     <a href="#">iPhone 5</a>
             </li>
     </ul>
</div>
<div class="show-separators f-right">
     <ul class="links">
             <li class="first">
                     <a href="#">Add</a>
             </li>
             <li>
                     <a href="#">More</a>
```

```
                </li>
                <li>
                        <a href="#">Custom</a>
                </li>
                <li class="last">
                        <a href="#">Links</a>
                </li>
        </ul>
</div>
<span class="section-line"></span>
```

Bottom Block

```
<div class="show-separators clearer" style="background-color: #eee; padding: 10px 15px;">

        <ul class="links">
                <li class="label">Featured Categories:</li>
                <li class="first">
                        <a href="{{store direct_url='fashion/tops.html'}}">Evening Tops</a>
                </li>
                <li>
                        <a href="#">Dresses</a>
                </li>
                <li class="last">
                        <a href="#">Shirts & Blouses</a>
                </li>
        </ul>

</div>
```

13.3 Custom links in the menu

The main menu typically includes the selection of categories from the catalog. But you can also add links to content pages and external websites. In order to add a link, follow the steps described in the example below.

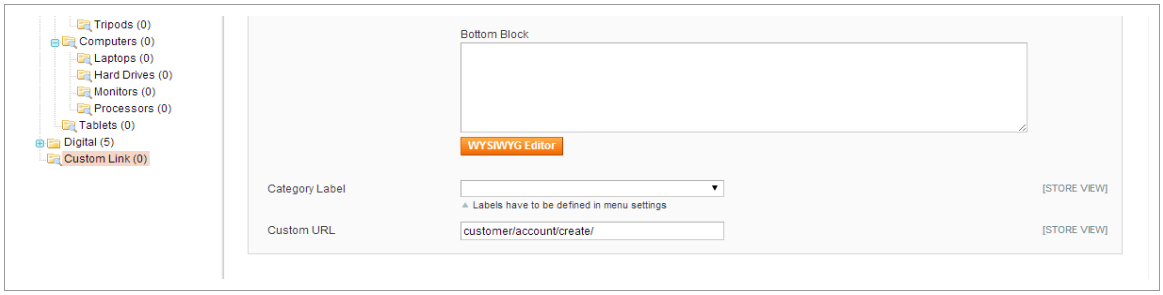
Let's say you want to create a link to Magento's “Create an Account” page.

1. Go to `Catalog > Category Management` and create a new category in your catalog by clicking the **Add Subcategory** button.

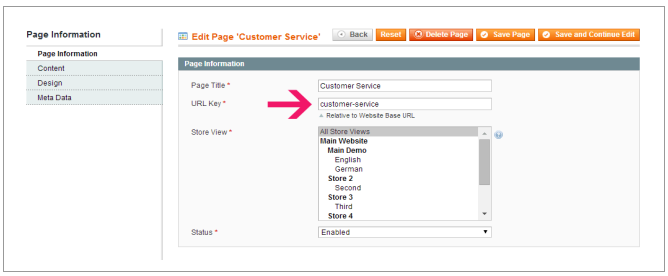
2. Set **Is Active** to “Yes”.

3. The name of the category will be the name of the link so you can use any name you like.

4. Open the `Menu` tab. In the **Custom URL** field, enter URL path of the “Create an Account” page which is `customer/account/create/`. This path will be appended to the base URL of the store to create a link to that page:



To create a link to some other page you just need to enter proper URL path. For example, to create a link to a CMS page, in the **Custom URL** field, enter the **URL Key** of that page. To check what is the **URL Key** of a page, go to CMS > Pages, select a page and find the field:



5. Finally, go to your store and make sure the new link appears in the main menu. If it's not, you may need to reindex Magento (**4.2 Index management**) and flush the cache (**4.1 Caching system**).

Alternatively, to add custom link to the menu, you can use the method (which uses URL rewrites) described in this article: *http://www.magentocommerce.com/knowledge-base/entry/adding-page-links-in-the-navigation-bar*

Custom links inside a static block

Another method to add custom links to the main menu, is to create a static block (in `CMS > Static Blocks`) with identifier **block_nav_links**. Content of this block will be displayed in the menu bar so you can quickly add multiple links to the menu.

To display links properly, content of the block should have the following structure:

```
<li class="nav-item level0 level-top right">
     <a class="level-top" href="{{store direct_url='about-magento-demo-store'}}">
            <span>Sample Internal Link</span>
     </a>
</li>
<li class="nav-item level0 level-top right">
     <a class="level-top" href="http://infortis-themes.com/">
            <span>Sample External Link</span>
     </a>
</li>
```

> **Note:**
>
> in version 1.9.0 a new class `nav-item` was added to the menu items `<li>`. You will need to update the content of your block and add that class in order to display links properly.

You can remove class `right` if you don't want the links to float to the right side of the menu bar.

Note that there are no `<ul>` tags at the beginning and at the end of this code snippet. This code will be inserted inside the menu bar HTML which already contains `<ul>` tags and additional tags would break the structure of the menu. Also, you may need to disable Magento's WYSIWYG editor, otherwise it will add `<ul>` tags automatically. It is recommended to disable the WYSIWYG completely so that it doesn't process the code

when you don't need it. Turn WYSIWYG on only when it is needed. Go to *System > Configuration > Content Management* and set it as `Disabled by Default`.

For more information about CMS **markup tags**, such as `{{store direct_url='about-magento-demo-store'}}`, refer to **4.8.3 CMS markup tags**.

13.4 Custom drop-down boxes in the menu

Drop-down boxes with custom content can be easily added to any top-level category in the main menu. In order to add a drop-down, follow the steps described in the example below.

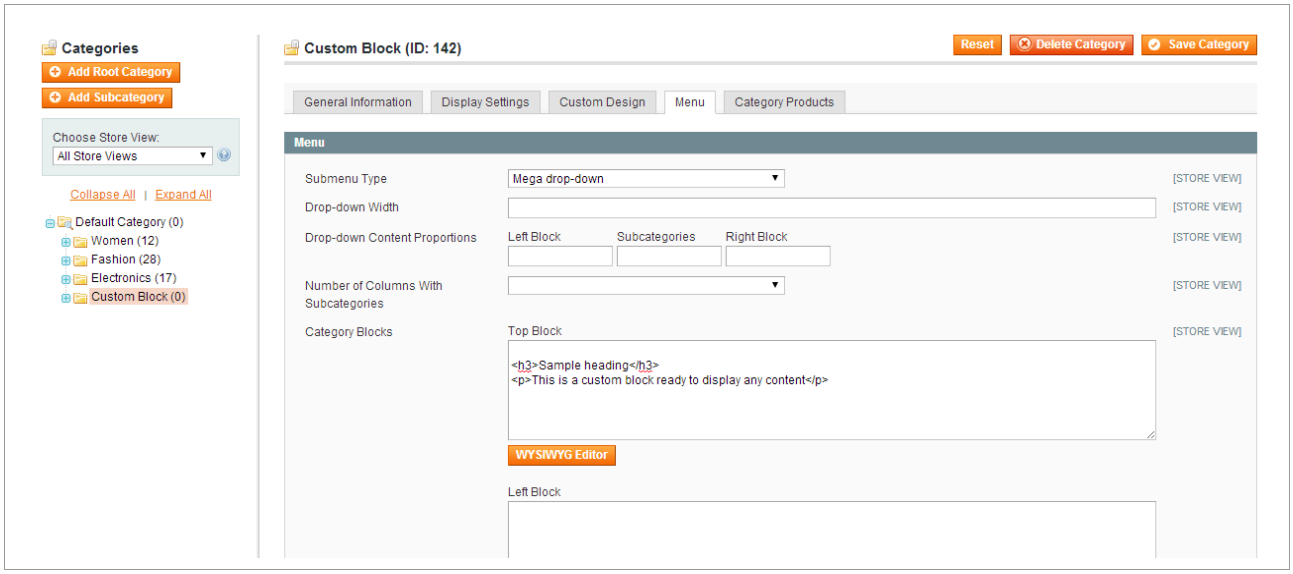
1. Go to `Catalog > Category Management` and create a new category in your catalog by clicking the **Add Subcategory** button.

2. Set **Is Active** to **Yes**.

3. The name of the category will be the name of the category item in the menu so you can use any name you like. Drop-down box will be displayed on mouse hover over that item.

4. Open the `Menu` tab. In the **Submenu Type** field, select “Mega drop-down”.

5. In the **Top Block** field, enter the content which you want to display in the drop-down box. It can be any HTML or just a simple text – whatever you need:



6. In the **Custom URL** field, enter a single hash character (`#`) to make this category not clickable (or leave this field empty if no changes are needed and the category in the menu has to be a link to the category page).

7. Finally, go to your store and make sure the new item appears in the menu. If it's not, you may need to reindex Magento (**4.2 Index management**) and flush the cache (**4.1 Caching system**).

Static block as a drop-down box

Another method to add custom drop-down box to the main menu, is to create a static block (in `CMS > Static Blocks`) with identifier `block_header_nav_dropdown`. Content of this block will be displayed in the menu as a drop-down box. Title of that block will be used as the name of the menu item.

To align this menu item to the right side of the menu bar, use option **Custom Static Block - Align Right** which can be found in `System > Configuration > Menu > Menu bar`.

14. Troubleshooting

14.1 General issues after installation

14.1.1 Some features don't work or give errors

Here's the list of possible sources of problems:

a) Magento cache (or external cache) was not flushed after installation of the theme.

b) CSS files which are responsible for the visual appearance of the theme were not generated automatically. Go to `System > Configuration > Ultimo > Theme Design` and click **Save Config** button. Then go to `System > Configuration > Ultimo > Theme Layout` and click **Save Config** button again. Then, flush Magento cache and refresh your browser's cache.

c) You can see all sorts of error messages if the **Compilation** mode was not disabled before installation.

d) Some of the theme files were not uploaded or are broken. Re-upload original theme files to your server (overwrite existing files) and flush the cache. Note that files can get broken if you send them in large numbers via FTP without zipping. So it is highly recommended to upload zipped files.

e) File permissions issue – refer to **14.1.4 File permissions** for more details.

f) Theme wasn't enabled properly. Theme should only be enabled in `System > Configuration > Design` (the same as it was described in the installation steps). Refer to **4.3 How to enable the theme** for more details.

g) You have some third-party extensions which change/break theme's default behavior or cause some conflicts. Disable all extensions and flush the cache, then enable extensions one by one to check which one is causing problems. Refer to **2.3 Third-party extensions** for more details about third-party extensions.

h) You edited theme files and accidentally removed or changed something in the files. Re-upload original theme files to your server (overwrite existing files) and flush the cache.

i) You have customized the theme by creating custom sub-theme and your

modifications override or break the default behavior of the theme.

j) You have some elements that left from your previous theme (e.g. in Magento's "base" theme in `app/design/frontend/base/`) and that breaks the new theme.

Also check your log files in `var/log/` (log settings should be enabled in *System > Config > Developer*) for more information about possible errors.

14.1.2 Blank page or 404 not found – when trying to open theme admin pages

Log out from Magento admin panel, delete all files in `var/cache/` directory (to clear Magento cache), and log in again.

14.1.3 There is no theme admin tab in the admin panel, or I get a blank page, or Access Denied error

Log out from Magento admin panel, delete all files in `var/cache/` directory, clear web browser cache/cookies for your store domain and log in again.

14.1.4 File permissions

Incorrect file permissions on your server may cause many issues. After uploading any files to your server in order to install themes or extensions you need to set correct file permissions for all the uploaded files – the same permissions as for the default Magento files.

File permissions may differ depending on server – usually 755 for folders and 644 for files. In case of any doubts about permissions, please contact your hosting provider. Refer to this article for more details:
http://www.magentocommerce.com/wiki/1_-_installation_and_configuration/magento_filesystem_permissions

In case of problems related to images refer to: **14.1.10 Images are not displayed**

14.1.5 Theme features don't work after installation or upgrade

If some theme features don't work after installation or some resources (e.g. images) are not displayed in the frontend, it often indicates problems with file permissions – your

Magento doesn’t have access to the theme files because of incorrect file permissions on your server. For more details refer to **14.1.4 File permissions**

After theme upgrade

If you upgraded the theme in your store and you're using custom sub-theme, you will also need to **upgrade your sub-theme**. This is extremely important step: review all the files in your sub-theme to make sure they don’t override new features added to the theme.

For example: if the product page template file was changed in the update (e.g. a new option was added: ability to display product SKU), you will need to apply that change in your sub-theme (in the product page template file). Otherwise SKU will not be displayed in your store because Magento will use the template file in your sub-theme to render the product page and your template file doesn't have the new feature. So you will need to add the new feature manually (copy the new code from the original template file to the file in your sub-theme).

14.1.6 Fatal error: Class 'Mage_Default_Helper_Data' not found

If you see the following error message:

```
Fatal error: Class 'Mage_Default_Helper_Data' not found in ***/app/Mage.php on line 546
```

(where `***` stands for the directory of Magento on your server) it indicates incorrect installation:

- Magento's **Compilation** mode was not disabled in *System > Tools > Compilation* before theme installation,
- or the theme wasn't enabled properly. Theme should only be enabled in *System > Configuration > Design* (the same as it was described in the installation steps). Refer to **4.3 How to enable the theme** for more details.

14.1.7 "Package with this name does not exist" message is displayed when trying to enable the theme

This message means that:

a) theme files were not uploaded,

b) or they are not in the correct place,

c) or there is a problem with file permissions on your server – refer to chapter **14.1.4**

File permissions for more details.

14.1.8 After theme installation the store gives blank page or an error

Log out from Magento admin panel. Change the owner of the uploaded files to the web server user and set correct file permissions for them. Delete all files in `var/cache/` directory (do not delete the directory) and log in again.

14.1.9 Tabs, sliders or other interactive elements don't work

This may be caused by jQuery conflict: one of your custom extensions is loading another version of jQuery. You need to remove that version of jQuery or remove that extension.

14.1.10 Images are not displayed

Sometimes after installation some images on custom CMS pages or inside static blocks (e.g. images in the slideshow) are not displayed. First, check if images were uploaded and if correct paths to those files can be found in the source code of your page (click `CTRL+U` in most browsers).

> **Note:**
>
> in fresh Magento installation, the default sample home page tries to load sample images which doesn't exist in this theme (so you will see blank rectangles instead of images). Go to *`CMS > Pages > Home page`*, remove the default content or replace it with your own content.

If all files were uploaded and paths are correct, it means that your Magento doesn't have access to those images because of incorrect file permissions on your server. You need to set correct file permissions for all images and folders containing images (in most cases images are stored in Magento's `media` directory). Refer to chapter **14.1.4 File permissions** for more details.

14.1.11 Sliders: previous/next buttons don't work in some browsers

Please see this thread on Flexslider forum (Flexslider is a jQuery plugin used for sliders), some users reported that the issue can be caused by a Wacom Driver, they were able to

resolve the issue by unplugging their tablets:
https://github.com/woothemes/FlexSlider/issues/589#issuecomment-14804424

You may also want to reinstall your web browser and remove all plugins to check if some of them can cause conflicts.

14.1.12 Slideshow is not displayed on the home page

Make sure the configuration of the slideshow is correct. Enable all the static blocks which are used as slides in the slideshow. Finally, flush the cache.

If this doesn't help:

- Non-standard value in *System > Configuration > Web > Default Pages > Default Web URL*. Set the default value: `cms`

- Some custom extensions change the default behavior of Magento's home page, so the slideshow module is not able to load on the home page anymore. How to recognize that case: open the source code of the home page in your web browser (`CTRL+U` in most browsers). The `<body>` tag on the home page should have CSS class `cms-index-index` (this is Magento's default class). If you don't see this class, then it is probably that case, and you will need to disable the extension which is changing the default behavior of Magento's home page.

14.2 Configuration

14.2.1 Theme configuration changes do not appear on the frontend

Flush Magento cache in *System > Cache Management*, clear web browser cache/cookies for your store domain and refresh the page in your web browser.

> **Note:**
>
> if you created a sub-theme to customize the theme, do not copy folder `skin/frontend/<ThemeName>/default/css/_config/` (and files stored inside) to your sub-theme. If you do this, files inside this folder will override your theme settings.

14.2.2 "Failed generating CSS file" message after saving theme configuration

What can cause the problem:

a) You didn't upload all theme files to the following directory:
`app/design/frontend/base/default/template/infortis/ultimo/css`

(you should see there three files: `design.phtml`, `grid.phtml`, `layout.phtml`)

Or your Magento doesn't have access to those files because of incorrect file permissions on your server. Refer to chapter **14.1.4 File permissions** for more details.

b) Your Magento doesn't have permissions to write in the following directory:
`skin/frontend/ultimo/default/css/_config`

Set correct file permissions for that directory. Refer to chapter **14.1.4 File permissions** for more details.

14.3 Third-party extensions

14.3.1 One of my extensions doesn't work with this theme

This theme may not be compatible with some third-party extensions. It's impossible for theme author to make the theme compatible out of the box with all existing extensions because there are thousands of available Magento extensions.

Generally all extensions can work with all themes but sometimes they have to be integrated manually. That's because sometimes the extension and the theme override the same Magento template file so there can be a conflict. In such cases you need to customize the theme and merge two conflicting template files into one template file.

This isn't something specific to our themes, this is general rule for all Magento themes and extensions.

Refer to chapter **2.3 Third-party extensions** for more details.

14.3.2 One of the features stopped working after third-party extension was installed

There can be a conflict, if the extension and the theme override the same Magento template file. You should contact author of the extension for more information. Also, check your log files in `var/log/` (log settings should be enabled in *System > Config > Developer*) for more information about the error. Refer to chapter **2.3 Third-party extensions** for more details about third-party extensions.